NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

Iterative Methods for Estimation of 2-D AR Parameters
using a
Data-Adaptive Toeplitz Approximation Algorithm

by

John C. Eremic

September 1991

Thesis Advisor: Murali Tummala

Approved for public release; distribution is unlimited

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Naval Postgraduate School 6b. OFFICE SYMBOL (If applicable) Code 32	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School
6c. ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000	7b. ADDRESS (City, State, and ZIP Code) Monterey, California 93943-5000
8a. NAME OF FUNDING / SPONSORING ORGANIZATION 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. / WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification) Iterative Methods for Estimation of 2-D AR Parameters using a Data-Adaptive Toeplitz Approximation Algorithm

12. PERSONAL AUTHOR(S) John C. Eremic

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Master's Thesis	FROM _____ TO _____	1991 September	63

16. SUPPLEMENTARY NOTATION The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Toepltiz Approximation, Autoregressive Spectral Estimation Image Restoration

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

A new two-dimensional data-adaptive algorithm utilizing the iterative Toeplitz approximation method is presented. This algorithm provides a robust and efficient means for accurate estimation of 2-D autoregressive parameters representing spatially variant data arrays. Its convergence performance is comparable to that of the 2-D Recursive Least Squares (RLS) algorithm but it is computationally more efficient. The saving in computation is realized by reducing the size of the matrix to be inverted when solving the AR model normal equations. The ability of the algorithm to accurately estimate the model parameters using very small data sets and various windowing schemes are evaluated. Spectral estimates are compared for quarter-plane (QP) nonsymmetric half-plane (NSHP) and combined-quadrant (CQ) regions of support. Additionally, the algorithm is tested in noise cancellation and line enhancement applications using image arrays. This algorithm may be implemented for data-adaptive image processing or coding and for applications requiring 2-D spectral estimation.

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL Tummala, Murali	22b. TELEPHONE (Include Area Code) 408-646-2645 22c. OFFICE SYMBOL Code EC/Tu

DD Form 1473, JUN 86 *Previous editions are obsolete.* SECURITY CLASSIFICATION OF THIS PAGE

S/N 0102-LF-014-6603

Approved for public release; distribution is unlimited

Iterative Methods for Estimation of 2–D AR Parameters
using a
Data-Adaptive Toeplitz Approximation Algorithm

by

John C. Eremic
Lieutenant, USN
B.S., University of the State of New York, 1983

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL

September, 1991

Michael A. Morgan, Chairman
Department of Electrical and Computer Engineering

ABSTRACT

A new two-dimensional data-adaptive algorithm utilizing the iterative Toeplitz approximation method is presented. This algorithm provides a robust and efficient means for accurate estimation of 2–D autoregressive parameters representing spatially variant data arrays. Its convergence performance is comparable to that of the 2–D Recursive Least Squares (RLS) algorithm but is computationally more efficient. The savings in computation is realized by reducing the size of the matrix to be inverted when solving the AR model normal equations. The ability of the algorithm to accurately estimate the model parameters using very small data sets and various windowing schemes are evaluated. Spectral estimates are compared for quarter-plane (QP), nonsymmetric half-plane (NSHP) and combined-quadrant (CQ) regions of support. Additionally, the algorithm is tested in noise cancellation and line enhancement applications using image arrays. This algorithm may be implemented for data-adaptive image processing or coding and for applications requiring 2–D spectral estimation.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENT

The successful completion of this thesis would not have been possible without the inspiration, guidance and support of many individuals. In particular, I must express my sincere gratitude to my thesis advisor and mentor, Dr. Murali Tummala. Additionally, I owe thanks to Professor Charles W. Therrien, my second reader, and the faculty of the Electrical and Computer Engineering Department whose advice and guidance have proven to be invaluable throughout my graduate education. Finally, I must express my heartfelt appreciation to my wife, Linda, our children, Bruce and Keisha and my parents, Mildred and Melvin Eremic whose unwavering love and support have sustained me throughout my efforts.

I. INTRODUCTION

A. PARAMETER ESTIMATION OF DATA-ADAPTIVE ARRAYS

Two-dimensional (2–D) autoregressive (AR) modeling provides a means for the spectral estimation of signals received by an array of spatially distributed sensors. Additionally, the 2–D AR model parameters may be used to represent the original signal in a compact manner resulting in compression of the data to be processed or transmitted. This is particularly useful in many image processing applications. The traditional least squares or Wiener filtering based 2–D AR parameter estimation [Ref. 1] provide accurate spectral estimates but require the the inversion of a relatively large autocorrelation matrix. This is undesirable since matrix inversion is computationally expensive. Iterative methods, using a Toeplitz approximation algorithm, have been suggested [Ref. 2, 3] as an alternative to the direct Wiener filter solution. The iterative Toeplitz approximation method has proven to be an effective means to estimate the parameters of fixed data arrays without having to invert the correlation matrix. This method is particularly useful when data arrays of limited size must be processed.

The results obtained using iterative methods to obtain the AR parameters of fixed data arrays suggest that these methods may be used to some advantage when estimating parameters adaptively. Adaptive parameter estimation is necessary in many digital signal processing applications where the data is non-stationary. In these applications, real-time or near real-time processing is often desirable. This requires that any algorithm used must be computationally efficient and be capable of providing accurate estimates obtained from a minimum of data. This thesis addresses the implementation of the iterative Toeplitz approximation method for 2–D parameter

estimation of non-stationary data. It is shown that this algorithm can be efficiently implemented to obtain accurate estimates from very small data arrays.

B. EXPLANATION OF NOTATION

All rectangular matrices are denoted by a boldface, captial letter, e.g., $\mathbf{R}$. Column vectors are designated by a boldface lowercase letter, for example: $\mathbf{x}$. Occasionally, a vector created temporarily for the development of a mathematical expression will be represented by a lowercase, boldface, greek letter. An example of this would be the vector $\boldsymbol{\alpha}$. A special operator matrix will be represented by a calligraphic capital letter, such as $\mathcal{F}$. Scalar values are generally represented non-boldface lowercase arabic or greek letters, e.g., a or λ. Captital letters such as A or P are used to represent a variety of values or expressions. These include anything from a polynomial to the dimension of a matrix. The symbol T is reserved to indicate the transpose of a vector or matrix.

C. THESIS OUTLINE

The following describes the organization of the remainder this thesis. Chapter II serves as an introduction to the problem of 2–D AR modeling and parameter estimation. Particular attention is given to the formulation of the normal equations for quarter-plane (QP) and nonsymmetric half-plane (NSHP) regions of support. This chapter concludes with a description of the least squares algorithm and direct inversion method of solving the normal equations. This serves as the foundation for Chapter III which extends the concept of solving normal equations using the iterative Toeplitz approximation method to derive parameter estimates referred to as the fixed data method. Chapter IV proceeds with the development of the data-adaptive algorithm using Toeplitz approximation. Two-dimensional AR spectral estimation is discussed in Chapter V, along with a comparison of results using both fixed and

adaptive iterative methods for spectral estimation. Additionally, a discussion of the use of the data-adaptive algorithm for noise cancellation and line enhancement of image arrays is provided in Chapter VI, with experimental results. Conclusions and recommendations for future work are found in Chapter VII.

II. 2–D AUTOREGRESSIVE MODELING

A. OVERVIEW

This procedure assumes a stationary (homogeneous) random process $x[n_1, n_2]$ that is the response of an AR model excited by a white noise input $w[n_1, n_2]$ having a variance σ_w^2. The AR model as shown in Figure 2.1 may be described as an *all pole*

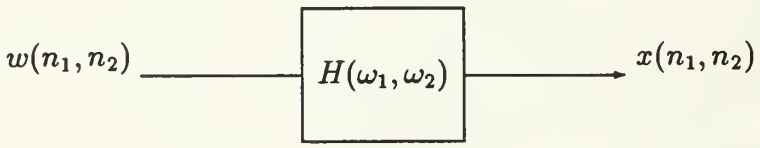

Figure 2.1: AR model excited by white noise.

filter with the transfer function

$$H(\omega_1, \omega_2) = \frac{1}{1 + \sum\sum_{(k_1,k_2 \in A)} a(k_1, k_2) e^{-j(\omega_1 k_1 + \omega_2 k_2)}} \tag{2.1}$$

where A is the region of support over which the parameters $a(k_1, k_2)$ are non-zero. The difference equation for the system that generates $x[n_1, n_2]$ can be written as

$$x[n_1, n_2] = -\sum_{(i,j)\ \in A}\sum a(i,j) x[n_1 - i, n_2 - j] + w[n_1, n_2]. \tag{2.2}$$

A linear set of equations for the filter coefficients $a(i, j)$ may be formed by multiplying both sides of (2.2) by $x[n_1 - l_1, n_2 - l_2]$ and computing the the statistical expectation of the resulting expression [Ref. 4]. This leads to the following expression called the *normal* equation

$$R_x[l_1, l_2] = -\sum_{(i,j)\ \in A}\sum a(i,j) R_x[l_1 - i, l_2 - j]\ , \tag{2.3}$$

for $l_1, l_2 \geq 0$. The coefficients $a(i,j)$ can be derived from this normal equation. It should be noted that the structure of the normal equation depends on the region of support A. A region of support may have any shape, but the most commonly used are the *quarter-plane* and *nonsymmetric half-plane.*

1. Quarter-Plane Support

The most straightforward region of support is the quarter-plane. A region A is considered to have quarter-plane (QP) support when $a(i,j)$ has non-zero values in one quadrant only as shown in Figure 2.2. For this case the normal equation has

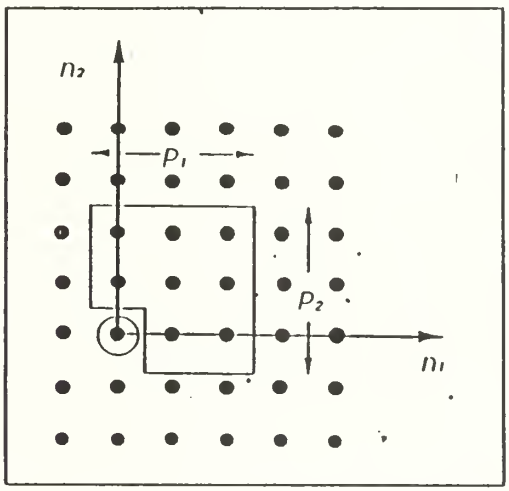

Figure 2.2: The quarter-plane region of support.

the form

$$R_x[l_1,l_2] = \sum_{i=0}^{P_1-1}\sum_{j=0}^{P_2-1} a(i,j)R_x[l_1-i,l_2-j],\ (i,j) \neq (0,0) \tag{2.4}$$

where $l_1 = 1,2,\ldots,P_1-1$ and $l_2 = 1,2,\ldots,P_2-1$ with P_1 and P_2 being the dimensions of A. If it is assumed that $a(0,0) = 1$, we may express (2.4) in *block matrix* form as

$$\begin{bmatrix} \mathbf{R}_0 & \mathbf{R}_{-1} & \mathbf{R}_{-2} & \cdots & \mathbf{R}_{-P_1+1} \\ \mathbf{R}_1 & \mathbf{R}_0 & \mathbf{R}_{-1} & \cdots & \mathbf{R}_{-P_1+2} \\ \mathbf{R}_2 & \mathbf{R}_1 & \mathbf{R}_0 & \cdots & \mathbf{R}_{-P_1+3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{R}_{P_1-1} & \mathbf{R}_{P_1-2} & \mathbf{R}_{P_1-3} & \cdots & \mathbf{R}_0 \end{bmatrix} \begin{bmatrix} \mathbf{a}_0 \\ \mathbf{a}_1 \\ \mathbf{a}_2 \\ \vdots \\ \mathbf{a}_{P_1-1} \end{bmatrix} = \begin{bmatrix} \mathcal{E}^{(0)} \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}. \tag{2.5}$$

Each block $\mathbf{R}_M$ of this matrix is given by

$$\mathbf{R}_M = \mathbf{R}_{-M}^T = \begin{bmatrix} R(M,0) & R(M,-1) & \cdots & R(M,-P_2+1) \\ R(M,1) & R(M,0) & \cdots & R(M,-P_2+2) \\ \vdots & \vdots & \ddots & \vdots \\ R(M,P_2-1) & R(M,P_2-2) & \cdots & R(M,0) \end{bmatrix} \tag{2.6}$$

while the coefficients are $\mathbf{a}_M = [a(M,0)\, a(M,1)\, a(M,2) \ldots a(M,P_2-1)]^T$ and the error variance vector is $\mathcal{E}^{(0)} = [\sigma_\epsilon^2 0 \ldots 0]^T$. The blocks $\mathbf{R}_M$ along the diagonals of the autocorrelation matrix $\mathbf{R}$ are equal and the diagonal elements of $\mathbf{R}_M$ are equal, thus $\mathbf{R}$ is *Toeplitz block-Toeplitz.*

2. Nonsymmetric Half-Plane Support

A region A with non-zero $a(i,j)$ as shown in Figure 2.3 is considered to have nonsymmetric half-plane (NSHP) support. The normal equation for this case

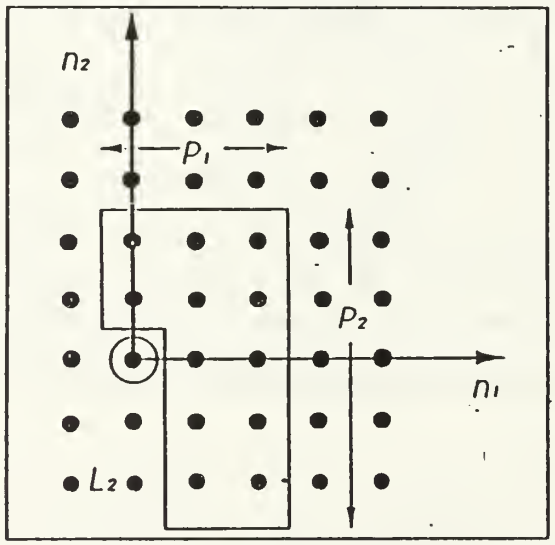

Figure 2.3: The nonsymmetric half-plane region of support.

must be modified to account for non-zero $a(0,j)$ in the fourth quadrant. This may be written as

$$R_x[l_1,l_2] = \sum_{j=1}^{L_2+P_2-1} a(0,j) R_x[l_1, l_2-j] + \sum_{i=0}^{P_1-1} \sum_{j=0}^{P_2-1} a(i,j) R_x[l_1-i, l_2-j]\,, \tag{2.7}$$

where $(i,j) \neq 0$, $l_1 = 0, 1, 2, \ldots, P_1 - 1$ and

$$l_2 = \begin{cases} 0, 1, 2, \ldots, L_2 + P_2 - 1, & \text{for } l_1 = 0 \\ L_2, L_2 + 1, L_2 + 2, \ldots, L_2 + P_2 - 1, & \text{for } l_1 \neq 0 \end{cases}.$$

The dimensions of A are given by P_1 and P_2 and L_2 is a negative number as defined in Figure 2.3. As in the quarter-plane case, if we let $a(0,0) = 1$ we may write (2.7) in block matrix form as

$$\begin{bmatrix} \tilde{\mathbf{R}}_0 & \tilde{\mathbf{R}}_{-1} & \tilde{\mathbf{R}}_{-2} & \cdots & \tilde{\mathbf{R}}_{-P_1+1} \\ \tilde{\mathbf{R}}_1 & \mathbf{R}_0 & \mathbf{R}_{-1} & \cdots & \mathbf{R}_{-P_1+2} \\ \tilde{\mathbf{R}}_2 & \mathbf{R}_1 & \mathbf{R}_0 & \cdots & \mathbf{R}_{-P_1+3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \tilde{\mathbf{R}}_{P_1-1} & \mathbf{R}_{P_1-2} & \mathbf{R}_{P_1-3} & \cdots & \mathbf{R}_0 \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{a}}_0 \\ \mathbf{a}_1 \\ \mathbf{a}_2 \\ \vdots \\ \mathbf{a}_{P_1-1} \end{bmatrix} = \begin{bmatrix} \tilde{\mathcal{E}}^{(0)} \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix}. \tag{2.8}$$

The blocks $\tilde{\mathbf{R}}_0$, $\tilde{\mathbf{R}}_M$ and $\mathbf{R}_M$ each have different forms. The diagonal block $\tilde{\mathbf{R}}_0$ has the form

$$\tilde{\mathbf{R}}_0 = \begin{bmatrix} R(0,0) & R(0,-1) & \cdots & R(0,-L_2-P_2+1) \\ R(0,1) & R(0,0) & \cdots & R(0,-L_2-P_2+2) \\ \vdots & \vdots & \ddots & \vdots \\ R(0,l_2+P_2-1) & R(0,L_2+P_2-2) & \cdots & R(0,0) \end{bmatrix}, \tag{2.9}$$

the blocks $\tilde{\mathbf{R}}_M = \tilde{\mathbf{R}}_{-M}^T$ in the first row and column may be written as

$$\begin{bmatrix} R(M,-L_2) & R(M,-L_2-1) & \cdots & R(M,-L_2-P_2+1) \\ R(M,-L_2+1) & R(M,-L_2) & \cdots & R(M,-L_2-P_2+2) \\ \vdots & \vdots & \ddots & \vdots \\ R(M,-L_2+P_2-1) & R(M,-L_2+P_2-2) & \cdots & R(M,0) \end{bmatrix}, \tag{2.10}$$

and the remaining blocks $\mathbf{R}_M$ are given by (2.6).

The model parameter vectors are given by

$$\tilde{\mathbf{a}}_0 = [a(0,0)\, a(0,1)\, a(0,2) \ldots a(0,L_2+P_2-1)]^T$$

and

$$\mathbf{a}_M = [a(M,L_2)\, a(M,L_2+1)\, a(M,L_2+2) \ldots a(M,L_2+P_2-1)]^T$$

for $M = 1, 2, \ldots, P_1 - 1$. The error variance vector $\tilde{\mathcal{E}}^{(0)} = [\sigma_\epsilon^2 0 \ldots 0]^T$. Except for the upper diagonal block (2.9) and top and left most blocks (2.10), the NSHP autocorrelation matrix is block-toeplitz with each block being toeplitz as well. Quarter-Plane support can be considered a special case of NSHP support with $L_2 = 0$.

B. SOLVING THE NORMAL EQUATIONS BY DIRECT INVERSION

The normal equations also arise in the 2–D linear prediction problem. When solving the normal equations, the objective is to find the parameters which minimize the prediction error [Ref. 5]. It is frequently more convenient to describe the formulation of the normal equations in terms of linear prediction.

In the 2–D least squares problem, the error between the random process $x[n_1, n_2]$ and its estimate $\hat{x}[n_1, n_2]$ is given by

$$e[n_1, n_2] = x[n_1, n_2] - \hat{x}[n_1, n_2] \tag{2.11}$$

or in expanded form as

$$e[n_1, n_2] = x[n_1, n_2] + \sum_{(i,j)\ \in A} \sum a(i, j)x[n_1 - i, n_2 - j],\ (i, j) \neq (0, 0). \tag{2.12}$$

The objective of the least squares method is to minimize the squared errors from a particular set of these terms [Ref. 4]. If we let $a(0, 0) = 1$, we can express (2.12) as

$$e[n_1, n_2] = \sum_{(i,j)\ \in A} \sum a(i, j)x[n_1 - i, n_2 - j] \tag{2.13}$$

which is compactly represented in vector notation as $\mathbf{e} = \mathbf{Xa}$. The error $e[n_1, n_2]$ is computed for each position of the filter, then normal equations are formed by multiplying both sides of (2.13) by $\mathbf{X}^T$ and applying the orthogonality principle [Ref. 5]. This results in

$$\mathbf{Ra} = \mathcal{E} \tag{2.14}$$

where $\mathbf{R} = \mathbf{X}^T\mathbf{X}$ and the error variance vector $\mathcal{E} = \mathbf{X}^T\mathbf{e} = [\mathcal{E}^{(0)T}, 0, \ldots, 0]^T$ with $\mathcal{E}^{(0)} = [\sigma_\epsilon^2, 0, \ldots, 0]^T$. The matrix $\mathbf{X}$ may be defined as

$$\mathbf{X} = \begin{bmatrix} | & \\ \mathbf{x}_0 & \mathbf{X}' \\ | & \end{bmatrix} = \begin{bmatrix} | & | & & | \\ \mathbf{x}_0 & \mathbf{x}_1 & \cdots & \mathbf{x}_{P_1-1} \\ | & | & & | \end{bmatrix} \tag{2.15}$$

and the model parameters may be given by $\mathbf{a} = [1 \quad \mathbf{a}']^T$. The parameter estimates are obtained from (2.14) by muliplying both sides of the equation by the inverse of $\mathbf{R}$ which may be written in expanded form as

$$\mathbf{a}' = -(\mathbf{X}'^T\mathbf{X}')^{-1}\mathbf{X}'^T\mathbf{x}_0 \,. \tag{2.16}$$

This is referred to as the *direct inversion method.*

The preceding discussion assumes that $\mathbf{R}$ is Toeplitz block-Toeplitz. This true for the case where the *autocorrelation method* [Ref. 5] is used to form the correlation matrix or when the theoretical correlation is known. In practicality, however, parameter estimates must be obtained from relatively small sets of finite data. In this case, it is often desirable to compute $\mathbf{R}$ using the *covariance method.* This results in a block autocorrelation matrix $\mathbf{R}$ with non-Toeplitz blocks $\mathbf{R}_M$ [Ref. 2]. The rows of the matrix $\mathbf{X}$ are the elements of the random process covered by the filter mask for each point being estimated as the mask is moved over the data. In the covariance method the filter mask is not moved past the boundaries of the region of support. This means that when QP support is used, $\mathbf{X}$ will be of dimension $P_1P_2 \times P_1P_2$ where P_1 and P_2 are the dimensions of the data array accessible to the filter mask. For example, using a 3×3 (nine element) filter mask to form the normal equations of an 8×8 data array would result in an $\mathbf{X}$ matrix with dimension 36×9. It should be noted that $P_1 = P_2 = 6$, since two rows and two columns of data cannot be estimated by the filter.

The preceding discussion provides the basis for the subsequent work in this

thesis. The next chapter will addresses this same problem using the iterative Toeplitz approximation method.

III. ITERATIVE TOEPLITZ APPROXIMATION ALGORITHM

The major drawback of the least squares method described in the previous chapter comes from the requirement to invert a relatively large autocorrelation matrix. This is undesireable since matrix inversion is computationally expensive. In this chapter we present an iterative method for solving the normal equations which does not involve the direct inversion of the correlation matrix to obtain the model parameters. This method is then extended to the case where the covariance method is used to form the correlation matrix and the normal equations are solved iteratively using Toeplitz approximation.

A. THE ITERATIVE SOLUTION OF THE NORMAL EQUATIONS

An alternative to direct inversion is to take advantage of the near Toeplitz-block-Toeplitz structure of the autocorrelation matrix and successively partition the normal equation [Ref. 2]. This partitioning permits the estimation of parameters while requiring the inversion of a matrix that is significantly smaller than the original autocorrelation matrix. The following discussion develops the iterative solution for QP and NSHP regions of support.

1. Quarter-Plane Support

In order to iteratively solve the QP normal equations we must begin by dividing both sides of (2.5) by σ^2. This results in a modified normal equation given

by

$$\begin{bmatrix} \mathbf{R}_0 & \mathbf{R}_{-1} & \mathbf{R}_{-2} & \cdots & \mathbf{R}_{-P_1+1} \\ \mathbf{R}_1 & \mathbf{R}_0 & \mathbf{R}_{-1} & \cdots & \mathbf{R}_{-P_1+2} \\ \mathbf{R}_2 & \mathbf{R}_1 & \mathbf{R}_0 & \cdots & \mathbf{R}_{-P_1+3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{R}_{P_1-1} & \mathbf{R}_{P_1-2} & \mathbf{R}_{P_1-3} & \cdots & \mathbf{R}_0 \end{bmatrix} \begin{bmatrix} \mathbf{a}_0'' \\ \mathbf{a}_1'' \\ \mathbf{a}_2'' \\ \vdots \\ \mathbf{a}_{P_1-1}'' \end{bmatrix} = \begin{bmatrix} \mathcal{E}^{(0)} \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix} . \tag{3.1}$$

where $\mathcal{E}^{(0)} = [1\,0\,0, \ldots, 0]^T$ and $\mathbf{a}_i'' = \frac{1}{\sigma^2}\mathbf{a}_i$ for $i = 1, 2, \ldots, P_1 - 1$. We now partition the normal equation (3.1) as follows

$$\begin{bmatrix} \mathbf{G}_1 & \mathbf{h}_1 \\ \mathbf{h}_1^T & \mathbf{R}_0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha}_1 \\ \mathbf{a}_{P-1}'' \end{bmatrix} = \begin{bmatrix} \boldsymbol{\gamma}_1 \\ \mathbf{0} \end{bmatrix} , \tag{3.2}$$

where

$$\mathbf{G}_1 = \begin{bmatrix} \mathbf{R}_0 & \mathbf{R}_{-1} & \mathbf{R}_{-2} & \cdots & \mathbf{R}_{-P_1+2} \\ \mathbf{R}_1 & \mathbf{R}_0 & \mathbf{R}_{-1} & \cdots & \mathbf{R}_{-P_1+3} \\ \mathbf{R}_2 & \mathbf{R}_1 & \mathbf{R}_0 & \cdots & \mathbf{R}_{-P_1+4} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{R}_{P_1-2} & \mathbf{R}_{P_1-3} & \mathbf{R}_{P_1-4} & \cdots & \mathbf{R}_0 \end{bmatrix} , \tag{3.3}$$

$$\mathbf{h}_1^T = [\mathbf{R}_{P_1-1} \quad \mathbf{R}_{P_1-2} \quad \mathbf{R}_{P_1-3} \quad \cdots \quad \mathbf{R}_1] , \tag{3.4}$$

$$\boldsymbol{\alpha}_1 = [\mathbf{a}_0''^T \quad \mathbf{a}_1''^T \quad \mathbf{a}_2''^T \quad \ldots \quad \mathbf{a}_{P_1-2}''^T]^T , \tag{3.5}$$

and

$$\boldsymbol{\gamma}_1 = [\mathcal{E}^{(0)T} \quad \mathbf{0}^T \quad \ldots \quad \mathbf{0}^T]^T . \tag{3.6}$$

We may now form a set of coupled iterative equations by defining an operator $\mathcal{F}_1$ which is given as

$$\mathcal{F}_1 = \begin{bmatrix} \mathbf{G}_1^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_0^{-1} \end{bmatrix} , \tag{3.7}$$

and using this operator to premultiply both sides of (3.2) to yield

$$\boldsymbol{\alpha}_1 = \mathbf{G}_1^{-1}[\boldsymbol{\gamma}_1 - \mathbf{h}_1\, \mathbf{a}_{P_1-1}''] , \tag{3.8}$$

and

$$\mathbf{a}_{P_1-1}'' = -\mathbf{R}_0^{-1}\mathbf{h}_1^T\boldsymbol{\alpha}_1 . \tag{3.9}$$

Equations (3.8) and (3.9) suggest an iterative solution and can be written as

$$\boldsymbol{\alpha}_1^{(k)} = \mathbf{G}_1^{-1}[\boldsymbol{\gamma}_{\ 1} - \mathbf{h}_1\, \mathbf{a}''^{(k-1)}_{P_1-1}] \,, \tag{3.10}$$

and

$$\mathbf{a}''^{(k)}_{P_1-1} = -\mathbf{R}_0^{-1}\mathbf{h}_1^T \boldsymbol{\alpha}_1^{(k-1)} \,. \tag{3.11}$$

which provide a means to solve the normal equation (3.1) iteratively.

It can be noted that the submatrix $\mathbf{G}_1$ is nearly the same dimension as the original correlation matrix. This means that the inversion of $\mathbf{G}_1$ provides little advantage, computationally, over the direct inversion method. It is clear, however, that further partitioning of $\mathbf{G}_1$ will decrease the required complexity. Partitioning of $\mathbf{G}_1$ yields

$$\begin{bmatrix} \mathbf{G}_2 & \mathbf{h}_2 \\ \mathbf{h}_2^T & \mathbf{R}_0 \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha}_{\,2} \\ \mathbf{a}''_{P-2} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\gamma}_{\ 2} \\ \mathbf{0} \end{bmatrix} \,, \tag{3.12}$$

where

$$\mathbf{G}_2 = \begin{bmatrix} \mathbf{R}_0 & \mathbf{R}_{-1} & \mathbf{R}_{-2} & \cdots & \mathbf{R}_{-P_1+3} \\ \mathbf{R}_1 & \mathbf{R}_0 & \mathbf{R}_{-1} & \cdots & \mathbf{R}_{-P_1+4} \\ \mathbf{R}_2 & \mathbf{R}_1 & \mathbf{R}_0 & \cdots & \mathbf{R}_{-P_1+5} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{R}_{P_1-3} & \mathbf{R}_{P_1-4} & \mathbf{R}_{P_1-5} & \cdots & \mathbf{R}_0 \end{bmatrix} \,, \tag{3.13}$$

$$\mathbf{h}_2^T = [\,\mathbf{R}_{P_1-2} \quad \mathbf{R}_{P_1-3} \quad \mathbf{R}_{P_1-4} \quad \cdots \quad \mathbf{R}_1\,] \,, \tag{3.14}$$

$$\boldsymbol{\alpha}_{\,2} = [\,\mathbf{a}_0''^T \quad \mathbf{a}_1''^T \quad \mathbf{a}_2''^T \quad \ldots \quad \mathbf{a}''^T_{P_1-3}\,]^T \,, \tag{3.15}$$

and

$$\boldsymbol{\gamma}_{\ 2} = [\,\mathcal{E}^{(0)T} \quad \mathbf{0}^T \quad \ldots \quad \mathbf{0}^T\,]^T \,. \tag{3.16}$$

As before, both sides of (3.12) are premultiplied by an operator defined by

$$\mathcal{F}_2 = \begin{bmatrix} \mathbf{G}_2^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_0^{-1} \end{bmatrix} \,, \tag{3.17}$$

which results in

$$\alpha_2 = \mathbf{G}_1^{-1}[\gamma_{\ 2} - \mathbf{h}_2\, \mathbf{a}''_{P_1-2}]\,, \tag{3.18}$$

and

$$\mathbf{a}''_{P_1-2} = -\mathbf{R}_0^{-1}\mathbf{h}_2^T\alpha_2\,. \tag{3.19}$$

Equations (3.18) and (3.19) may be solved iteratively in the same manner as (3.8) and (3.9). This solution is expressed as

$$\alpha_2^{(k)} = \mathbf{G}_2^{-1}[\gamma_{\ 1} - \mathbf{h}_2\, \mathbf{a}''^{(k-1)}_{P_1-2}]\,, \tag{3.20}$$

and

$$\mathbf{a}''^{(k)}_{P_1-2} = -\mathbf{R}_0^{-1}\mathbf{h}_2^T\alpha_2^{(k-1)}\,. \tag{3.21}$$

This partitioning process may be continued until $\mathbf{G}_{P_1-1} = \mathbf{R}_0$. This will result in a partitioned normal equation given by

$$\begin{bmatrix} \mathbf{G}_{P_1-1} & \mathbf{h}_{P_1-1} \\ \mathbf{h}_{P_1-1}^T & \mathbf{R}_0 \end{bmatrix} \begin{bmatrix} \alpha_{P_1-1} \\ \mathbf{a}''_{P-2} \end{bmatrix} = \begin{bmatrix} \gamma_{\ P_1-1} \\ \mathbf{0} \end{bmatrix}\,, \tag{3.22}$$

where $\mathbf{h}_{P_1-1}^T = \mathbf{R}_1$, $\alpha_{P_1-1} = \mathbf{a}''_0$ and $\gamma_{\ P_1-1} = \mathcal{E}^{(0)}$.

Premultiplication of both sides of (3.22) by an operator $\mathcal{F}_{P_1-1}$ will result in the iterative solution

$$\mathbf{a}''^{(k)}_0 = \mathbf{R}_0^{-1}\mathcal{E}^{(0)} - [\mathbf{R}_{-1}\mathbf{a}_1''^{(k-1)} + \ldots + \mathbf{R}_{-P_1+1}\mathbf{a}''^{(k-1)}_{P_1-1}]\,, \tag{3.23}$$

and

$$\mathbf{a}_1''^{(k)} = -\mathbf{R}_0^{-1}[\mathbf{R}_1\mathbf{a}_0''^{(k-1)} + \mathbf{R}_{-1}\mathbf{a}_2''^{(k-1)} + \ldots + \mathbf{R}_{-P_1+2}\mathbf{a}''^{(k-1)}_{P_1-1}]\,. \tag{3.24}$$

For this case the operator matrix is given by

$$\mathcal{F}_{P_1-1} = \begin{bmatrix} \mathbf{R}_0^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_0^{-1} \end{bmatrix}\,. \tag{3.25}$$

At this point in our development, we can combine the preceding iterative solutions from the successive levels of partitioning into a set of compact iterative summations

$$\mathbf{a}_0''^{(k)} = \mathbf{a}_0''^{(0)} - \mathbf{R}_0^{-1} \sum_{i=1}^{P_1-1} \mathbf{R}_{-i} \mathbf{a}_i''^{(k-1)} , \tag{3.26}$$

$$\mathbf{a}_j''^{(k)} = -\mathbf{R}_0^{-1} \sum_{i=0}^{P_1-1} \mathbf{R}_{j-i} \mathbf{a}_i''^{(k-1)} , (i \neq j) , \tag{3.27}$$

where $\mathbf{a}_0''^{(0)} = \mathbf{R}_0^{-1} \mathcal{E}^{(0)}$, $\mathbf{a}_j''^{(0)} = -\mathbf{R}_0^{-1} \mathbf{R}_j \mathbf{a}_0''^{(0)}$, and $j = 1, 2, \ldots, P_1 - 1$. The index of iteration is k and $\mathbf{a}_i''$ are the $P_2 \times 1$ parameter vectors. The solution of (3.26)-(3.27) requires $\mathcal{O}(P_1^2 P_2^2 K)$ multiplications, where K represents the number of iterations to converge to the true parameter values. An additional $P_1 P_2^2$ multiplications are required to compute the initial parameter estimates $\mathbf{a}_j''^{(0)}$ for $j = 1, 2, \ldots, P_2 - 1$. This results in total multplications of $\mathcal{O}(P_1^2 P_2^2 K + P_1 P_2^2)$. Depending on the number of iterations required to converge to the true parameters, this total compares with the algorithms of Wax and Kalaith [Ref. 6] and Akaike [Ref. 7] which require $\mathcal{O}(P_1^3 P_2^2)$ operations. When large arrays are used this represents a noticeable savings over direct inversion which requires $\mathcal{O}(P_1^3 P_2^3)$ multiplications.

2. Nonsymmetric Half-Plane Support

The derivation of the iterative NSHP solution follows a procedure of succesive partitioning similar to that of QP support. However some differences exist due to the asymmetry of the autocorrelation matrix consisting of three different types of block matrices. This asymmetry results in a somewhat modified set of recursive summations given by

$$\tilde{\mathbf{a}}_0''^{(k)} = \tilde{\mathbf{a}}_0''^{(0)} - \tilde{\mathbf{R}}_0^{-1} \sum_{i=1}^{P_1-1} \tilde{\mathbf{R}}_{-i} \mathbf{a}''^{(k-1)} \tag{3.28}$$

$$\mathbf{a}_j''^{(k)} = -\mathbf{R}_0^{-1} \tilde{\mathbf{R}}_0^{-1} \tilde{\mathbf{a}}_0''^{(k-1)} - \mathbf{R}_0^{-1} \sum_{i=1}^{P_1-1} \mathbf{R}_{j-i} \mathbf{a}''^{(k-1)}, \ (i \neq j), \tag{3.29}$$

where $\tilde{\mathbf{a}}_0''^{(0)} = \tilde{\mathbf{R}}_0^{-1}\tilde{\mathcal{E}}^{(0)}$, $\mathbf{a}_j''^{(0)} = -\tilde{\mathbf{R}}_0^{-1}\tilde{\mathbf{R}}_j\tilde{\mathbf{a}}_0''^{(0)}$, and $j = 1, 2, \ldots, P_1 - 1$. The index of iteration is k and $\mathbf{a}_i''$ are the $P_2 \times 1$ parameter vectors. As with QP support, The solution of (3.28)-(3.29) requires $\mathcal{O}(P_1^2 P_2^2 K)$ multiplications, where K represents the number of iterations to converge to the true parameter values.

B. THE ITERATIVE SOLUTION USING TOEPLITZ APPROXIMATION

The discussion of the previous section assumes an autocorrelation matrix with Toeplitz-block-Toeplitz structure. In most cases the autocorrelation matrix must be formed from a limited, and possibly very small, amount of sampled data. In this situation, it is best to use the covariance method to estimate the correlation matrix to reduce the bias in the estimate. However, since this method does not have Toeplitz-block-Toeplitz structure, it is necessary to modify the iterative algorithm described above.

1. Forward Iteration

For first quadrant QP support, the covariance estimate $\mathbf{R}$ has the form

$$\begin{bmatrix} \mathbf{R}_{0,0} & \mathbf{R}_{0,1} & \mathbf{R}_{0,2} & \cdots & \mathbf{R}_{0,P_1-1} \\ \mathbf{R}_{1,0} & \mathbf{R}_{1,1} & \mathbf{R}_{1,2} & \cdots & \mathbf{R}_{1,P_1-1} \\ \mathbf{R}_{2,0} & \mathbf{R}_{2,1} & \mathbf{R}_{2,2} & \cdots & \mathbf{R}_{2,P_1-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{R}_{P_1-1,0} & \mathbf{R}_{P_1-1,1} & \mathbf{R}_{P_1-1,2} & \cdots & \mathbf{R}_{P_1-1,P_1-1} \end{bmatrix}. \tag{3.30}$$

A proven procedure [Ref. 8] that can be used to form the Toeplitz approximation is to first average the diagonal blocks $\mathbf{R}_{j,j}$, $i = j$ and then form a Toeplitz $\mathbf{T}$ matrix from this averaged block $\mathbf{R}_{avg}$ by averaging its diagonal elements. The diagonal elements of $\mathbf{T}$ can be obtained as follows

$$t(i) = \frac{1}{P_2 - i} \sum_{j=0}^{P_2-i-1} \mathbf{R}_{avg}(i + j, j) \tag{3.31}$$

where $i = 0, 1, \ldots, P_2 - 1$.

We may now proceed to succesively partition the normal equations in the manner outlined in the previous section. The first partitioning leads to the iterative equations

$$\boldsymbol{\alpha}_1^{\prime\prime(k)} = \mathbf{G}_1^{-1}[\boldsymbol{\gamma}_{\ 1} - \mathbf{h}_1\, \mathbf{a}_{P_1-1}^{\prime\prime(k-1)}] \,, \tag{3.32}$$

and

$$\mathbf{a}_{P_1-1}^{\prime\prime(k)} = -\mathbf{R}_{P_1-1,P_1-1}^{-1}\mathbf{h}_1^T\boldsymbol{\alpha}_1^{(k-1)} \,. \tag{3.33}$$

The diagonal block $\mathbf{R}_{P_1-1,P_1-1}$ may be written as

$$\mathbf{R}_{P_1-1,P_1-1} = \mathbf{T} + \Delta_{P_1-1,P_1-1} \tag{3.34}$$

where Δ_{P_1-1,P_1-1} is the difference between $\mathbf{R}_{P_1-1,P_1-1}$ and the Toeplitz approximation $\mathbf{T}$.

Substituting (3.34) into (3.33) modifies the recursion to give

$$\boldsymbol{\alpha}_1^{(k)} = \mathbf{G}_1^{-1}[\boldsymbol{\gamma}_{\ 1} - \mathbf{h}_1\, \mathbf{a}_{P_1-1}^{\prime\prime(k-1)}] \,, \tag{3.35}$$

and

$$\mathbf{a}_{P_1-1}^{\prime\prime(k)} = -\mathbf{T}^{-1}\Delta_{P_1-1,P_1-1}\mathbf{a}_{P_1-1}^{\prime\prime(k-1)} - \mathbf{T}^{-1}\mathbf{h}_1^T\boldsymbol{\alpha}_1^{(k-1)} \,. \tag{3.36}$$

The final expressions resulting from the successive partitioning are then

$$\boldsymbol{\alpha}_{P_1-1}^{(k)} = \mathbf{G}_{P_1-1}^{-1}[\boldsymbol{\gamma}_{\ P_1-1} - \mathbf{h}_{P_1-1}\, \mathbf{a}_1^{\prime\prime(k-1)}] \,, \tag{3.37}$$

and

$$\mathbf{a}_1^{\prime\prime(k)} = -\mathbf{R}_{1,1}\mathbf{h}_{P_1-1}\boldsymbol{\alpha}_{P_1-1}^{(k-1)} \,. \tag{3.38}$$

where $\boldsymbol{\alpha}_{P_1-1} = \mathbf{a}_0, \mathbf{G}_{P_1-1} = \mathbf{R}_{0,0}$, and $\mathbf{h}_{P_1-1} = \mathbf{R}_{0,1}$. The diagonals $\mathbf{R}_{j,j}$ may be rewritten as

$$\mathbf{R}_{j,j} = \mathbf{T} + \Delta_{j,j} \,, \tag{3.39}$$

Now, substituting (3.39) into (3.38) results in

$$\alpha_{P_1-1}^{(k)} = \mathbf{G}_{P_1-1}^{-1}[\gamma_{P_1-1} - \mathbf{h}_{P_1-1}\, \mathbf{a}_1^{\prime\prime(k-1)}] \,, \tag{3.40}$$

and

$$\mathbf{a}_1^{\prime\prime(k)} = -\mathbf{T}^{-1}\Delta_{1,1}\mathbf{a}_1^{\prime\prime(k-1)} - \mathbf{T}^{-1}\mathbf{h}_1^T \alpha_{P_1-1}^{\prime\prime(k-1)} \,. \tag{3.41}$$

Finally, combining the succesive iterative equations results in our solution to the normal equations using the *forward* iterative algorithm for the QP case, which takes the form

$$\mathbf{a}_0^{\prime\prime(k)} = \mathbf{a}_0^{\prime\prime(0)} - \mathbf{T}^{-1}\Delta_{0,0}\mathbf{a}_0^{\prime\prime(k-1)} - \mathbf{T}^{-1}\sum_{i=1}^{P_1-1}\mathbf{R}_{0,i}\mathbf{a}_i^{\prime\prime(k-1)} \tag{3.42}$$

$$\mathbf{a}_j^{\prime\prime(k)} = -\mathbf{T}^{-1}\Delta_{j,j}\mathbf{a}_j^{\prime\prime(k-1)} - \mathbf{T}^{-1}\sum_{\substack{i=0 \\ i\neq j}}^{P_1-1}\mathbf{R}_{j,i}\mathbf{a}_i^{\prime\prime(k-1)} \tag{3.43}$$

where $\mathbf{a}_0^{\prime\prime(0)} = \mathbf{T}^{-1}\mathcal{E}^{(0)}$, $\mathbf{a}_j^{\prime\prime(0)} = \mathbf{T}^{-1}\mathbf{R}_{j,0}\mathbf{a}_0^{\prime\prime(0)}$ and $j = 1, 2, \ldots, P_1 - 1$. The forward iteration will provide parameters for the first quadrant spectral estimate.

2. Backward Iteration

A filter mask may have quarter-plane support in any two quadrants. A second quadrant estimate exists for the quadrant adjacent to either side of the quadrant designated to be the first quadrant. The Hermitian symmetry and Toeplitz-block-Toeplitz nature of the autocorrelation matrix causes the estimates produced from diagonally adjacent quadrants to be identical [Ref. 9].

The second quadrant AR parameters $b(i,j)$ may be estimated from the normal equations using the *backward* iterative Toeplitz approximation. When $b(0,0) = 1$ the second quadrant normal equation is given by

$$\begin{bmatrix} \mathbf{R}_{0,0} & \mathbf{R}_{0,1} & \mathbf{R}_{0,2} & \cdots & \mathbf{R}_{0,P_1-1} \\ \mathbf{R}_{1,0} & \mathbf{R}_{1,1} & \mathbf{R}_{1,2} & \cdots & \mathbf{R}_{1,P_1-1} \\ \mathbf{R}_{2,0} & \mathbf{R}_{2,1} & \mathbf{R}_{2,2} & \cdots & \mathbf{R}_{2,P_1-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{R}_{P_1-1,0} & \mathbf{R}_{P_1-1,1} & \mathbf{R}_{P_1-1,2} & \cdots & \mathbf{R}_{P_1-1,P_1-1} \end{bmatrix} \begin{bmatrix} \mathbf{b}_{P_1-1} \\ \mathbf{b}_{P_1-2} \\ \mathbf{b}_{P_1-3} \\ \vdots \\ \mathbf{b}_0 \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathcal{E}^{(0)} \end{bmatrix} . \tag{3.44}$$

where $\mathbf{R}_{i,j} = \mathbf{R}_{j,i}^T$, $\mathcal{E}^{(0)} = [1\,0\,0,\ldots,0]^T$ and

$$\mathbf{b}_k^T = (1/\sigma^2)[b(k,0), b(k,1), b(k,2), \ldots, b(k,P_1-1)] \,.$$

It may be noted that the second quadrant autocorrelation matrix is identical to that of the first quadrant. However, the parameter estimates $b(i,j)$ are not generally the same. The estimation of the backward parameters is computed iteratively in the same manner as the forward parameters, but, the backward method differs in the way that the normal equation is partitioned. In this case the partitioning begins at the upper left diagonal block and continues until the $\mathbf{G}$ matrix consists of only the lower right diagonal block. The Toeplitz approximation $\mathbf{T}$ is identical to that of first quadrant support.

The combined backward iterative equations, which are similar to those for the forward iteration, can be written as

$$\mathbf{b}_0^{(k)} = \mathbf{b}_0^{(0)} - \mathbf{T}^{-1}\Delta_{P_1-1,P_1-1}\mathbf{b}_0^{(k-1)} - \mathbf{T}^{-1}\sum_{i=1}^{P_1-1}\mathbf{R}_{P_1-1,P_1-1-i}\mathbf{b}_{P_1-1-i}^{(k-1)} \tag{3.45}$$

$$\mathbf{b}_j^{(k)} = -\mathbf{T}^{-1}\Delta_{P_1-1-j,P_1-1-j}\mathbf{b}_j^{(k-1)} - \mathbf{T}^{-1}\sum_{\substack{i=0\\ i\neq j}}^{P_1-1}\mathbf{R}_{P_1-1-j,P_1-1-i}\mathbf{b}_{P_1-1-i}^{(k-1)} \tag{3.46}$$

where $\mathbf{b}_0^{(0)} = \mathbf{T}^{-1}\mathcal{E}^{(0)}$, $\mathbf{b}_j^{(0)} = \mathbf{T}^{-1}\mathbf{R}_{P_1-1,j}\mathbf{b}_0^{(0)}$ and $j = 1,2,\ldots,P_1-1$. The first and second quadrant parameters are used to find the *combined-quadrant* spectral estimate, which is discussed in Chapter V.

3. Iterative Method for NSHP Support

The iterative Toeplitz approximation method may be used to estimate the parameters of arrays with nonsymmetric half-plane support. As shown previously the NSHP autocorrelation matrix has an asymmetric structure which can be expressed

as

$$\begin{bmatrix} \tilde{\mathbf{R}}_{0,0} & \tilde{\mathbf{R}}_{0,1} & \tilde{\mathbf{R}}_{0,2} & \cdots & \tilde{\mathbf{R}}_{0,P_1-1} \\ \tilde{\mathbf{R}}_{1,0} & \mathbf{R}_{1,1} & \mathbf{R}_{1,2} & \cdots & \mathbf{R}_{1,P_1-1} \\ \tilde{\mathbf{R}}_{2,0} & \mathbf{R}_{2,1} & \mathbf{R}_{2,2} & \cdots & \mathbf{R}_{2,P_1-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \tilde{\mathbf{R}}_{P_1-1,0} & \mathbf{R}_{P_1-1,1} & \mathbf{R}_{P_1-1,2} & \cdots & \mathbf{R}_{P_1-1,P_1-1} \end{bmatrix} \tag{3.47}$$

The dimensions of the top most and left most blocks are reduced by L_2 rows and columns, respectively. As in the previous cases the iterative process begins with forming the Toeplitz approximation of $\mathbf{R}_{avg}$. In the NSHP case, however, the upper left main diagonal block is not included in the average becuase it does not have the same dimension as the other main diagonal blocks. Thus, it is necessary to use the elements of the upper left main diagonal block to create a separate Toeplitz approximation designated as $\tilde{\mathbf{T}}$.

The normal equation is successively partitioned as described in the previous sections. At each stage of the partitioning, the blocks along the main diagonal are expressed as the sum of the Toeplitz approximation and a difference matrix. The iterative summations are given by

$$\tilde{\mathbf{a}}_0^{\prime\prime(k)} = \tilde{\mathbf{a}}_0^{\prime\prime(0)} - \tilde{\mathbf{T}}^{-1}\tilde{\Delta}_{0,0}\tilde{\mathbf{a}}_0^{\prime\prime(k-1)} - \tilde{\mathbf{T}}^{-1}\sum_{i=1}^{P_1-1}\mathbf{R}_{0,i}\mathbf{a}_i^{\prime\prime(k-1)} \tag{3.48}$$

$$\mathbf{a}_j^{\prime\prime(k)} = -\mathbf{T}^{-1}\tilde{\Delta}_{j,j}\mathbf{a}_j^{\prime\prime(k-1)} - \mathbf{T}^{-1}\tilde{\mathbf{R}}_{j,0}\tilde{\mathbf{a}}_0^{\prime\prime(k-1)} - \mathbf{T}^{-1}\sum_{\substack{i=0 \\ i\neq j}}^{P_1-1}\mathbf{R}_{j,i}\mathbf{a}_i^{\prime\prime(k-1)} \tag{3.49}$$

where $\tilde{\mathbf{a}}_0^{\prime\prime(0)} = \tilde{\mathbf{T}}^{-1}\tilde{\mathcal{E}}^{(0)}$, $\mathbf{a}_j^{\prime\prime(0)} = \mathbf{T}^{-1}\tilde{\mathbf{R}}_{j,0}\tilde{\mathbf{a}}_0^{\prime\prime(0)}$ and $j = 1, 2, \ldots, P_1 - 1$.

C. SUMMARY

It is clear that iterative methods have a computational advantage over the direct inversion method. This has been shown for the Toeplitz-block-Toeplitz case and extended to the case where Toeplitz approximation is used to compensate for the non-Toeplitz nature of autocorrelation matrices formed using the covariance method.

Generally, the matrix that must be inverted using the iterative method has the dimension of a single block in the block matrix. This corresponds to the filter order or mask size. The mask size will vary with the order necessary for accurate AR modeling of the sampled data, but filter masks that range from 3×3 to 6×6 will be sufficient for most applications. Although other methods may exhibit faster convergence to the true parameters, they do so with greater computational complexity. The iterative methods provide an approximation much sooner. Additionally, the iterative methods are guaranteed to converge when the block matrix is symmetric and positive definite [Ref. 2]. This property is important when extending the iterative Toeplitz approximation algorithm to the data-adaptive case which is introduced in the next chapter.

IV. DATA-ADAPTIVE ITERATIVE TOEPLITZ APPROXIMATION ALGORITHM

A. OVERVIEW

The necessity for adaptive filtering techniques arises when it is desired to process signals that result from an environment of unknown statistics [Ref. 10]. There exists a broad class of problems that fall into this category, these include such diverse fields as sonar, radar, image processing, seismology and communications. In general, adaptive filters provide a significant improvement in performance over fixed coefficient filters designed to operate on data with unknown statistics. Additionally, the use of adaptive filters makes possible new signal processing capabilties that would not be available otherwise.

One distinct advantage associated with adaptive filters pertains to their ability process data on-line. This is desirable for many applications such as autoregressive spectrum analysis, detection of a signal in noise, adaptive noise cancellation and line enhancement.

Two-dimensional adaptive filters are used to process data obtained from spatially variant data arrays. The most successful algorithms currently implemented for this purpose include the 2–D least mean square (LMS) and recursive least squares (RLS) algorithms. Both of these algorithms have been derived from the 2–D Wiener filter and method of least squares [Ref. 11]. With this in mind, it follows that the iterative method for fixed data may be successfully extended to operate on spatially variant arrays. The remainder of this chapter describes the development of the data-adaptive iterative Toeplitz approximation algorithms and some considerations that apply when using this algorithm.

B. THE DATA-ADAPTIVE ALGORITHM

The data-adaptive Toeplitz approximation algorithm is based on the same succesive partitioning described for the fixed data case. The algorithm becomes adaptive in nature when the autocorrelation matrix $\mathbf{R}_n$ is updated for each shift n of the filter mask. This yields a continuously updated set of AR parameter estimates. Computation of AR parameter estimates begins with the first iteration [Ref. 12]. As in the fixed data case, only the approximated Toeplitz matrix $\mathbf{T}$ is inverted. Computation time is shortened further by the fact that $\mathbf{R}_n$ is computed only from data covered by the filter mask, which amounts to taking the outer product of a $P_1P_2 \times 1$ vector where P_1P_2 is equal to the number of filter coefficients. The data-adaptive form of the combined iterative equations can be written as

$$\mathbf{a}_0''^{(n)} = \mathbf{a}_0''^{(0)} - \mathbf{T}_n^{-1}\Delta_{n,0,0}\mathbf{a}_0''^{(n-1)} - \mathbf{T}_n^{-1}\sum_{i=1}^{P_1-1}\mathbf{R}_{n,0,i}\mathbf{a}_i''^{(n-1)} \tag{4.1}$$

$$\mathbf{a}_j''^{(n)} = -\mathbf{T}_n^{-1}\Delta_{n,j,j}\mathbf{a}_j''^{(n-1)} - \mathbf{T}_n^{-1}\sum_{\substack{i=0\\ i\neq j}}^{P_1-1}\mathbf{R}_{n,j,i}\mathbf{a}_i''^{(n-1)} \tag{4.2}$$

where $\mathbf{a}_0''^{(0)} = \mathbf{T}_n^{-1}\mathcal{E}^{(0)}$, $\mathbf{a}_{\mathbf{o}j}''^{(0)} = \mathbf{T_o}^{-1}\mathbf{R}_{\mathbf{o}j,0}\mathbf{a}_0''^{(0)}$ and $j = 1, 2, \ldots, P_1 - 1$. These can be compared to (3.42) and (3.43) for the fixed data case. The initial parameter estimates $\mathbf{a_o}''$ are determined for the initial point being estimated and all subsequent parameters $\mathbf{a}_j''$ are a function of the parameters computed at the previous mask positions. The backward equations, used for the second quadrant estimates, can be written as

$$\mathbf{b}_0^{(n)} = \mathbf{b}_0^{(0)} - \mathbf{T}_n^{-1}\Delta_{n,P_1-1,P_1-1}\mathbf{b}_0^{(n-1)} - \mathbf{T}_n^{-1}\sum_{i=1}^{P_1-1}\mathbf{R}_{n,P_1-1,P_1-1-i}\mathbf{b}_{P_1-1-i}^{(n-1)} \tag{4.3}$$

$$\mathbf{b}_j^{(n)} = -\mathbf{T}_n^{-1}\Delta_{n,P_1-1-j,P_1-1-j}\mathbf{b}_j^{(n-1)} - \mathbf{T}_n^{-1}\sum_{\substack{i=0\\ i\neq j}}^{P_1-1}\mathbf{R}_{n,P_1-1-j,P_1-1-i}\mathbf{b}_{P_1-1-i}^{(n-1)} \tag{4.4}$$

where $\mathbf{b}_0^{(0)} = \mathbf{T}_n^{-1}\mathcal{E}^{(0)}$, $\mathbf{b}_{\mathbf{o}j}^{(0)} = \mathbf{T_o}^{-1}\mathbf{R}_{\mathbf{o}P_1-1,j}\mathbf{b}_0^{(0)}$ and $j = 1, 2, \ldots, P_1 - 1$.

It should be noted that the only difference between the data-adaptive equations and the fixed data equations is the the index of iteration which is n instead of k

and the manner in which the autocorrelation matrix is formulated. In the data adaptive case, the iteration proceeds with each shift of the filter mask or update of the autocorrelation matrix which has the recursive form

$$\mathbf{R}_n = \mathbf{R}_{n-1} + \mathbf{x}_n \mathbf{x}_n^T \,. \tag{4.5}$$

The $P_1 P_2 \times 1$ vector $\mathbf{x}_n$ is obtained by arranging the 2-D data covered by the filter mask at each shift $n = (n_1, n_2)$ in a vector form. The diagonal blocks of $\mathbf{R}_n$ are used to form $\mathbf{T}_n$ in the same manner as the fixed data case.

As with the RLS algorithm and other recursive implementations of the method of least squares, it is necessary to introduce a *weighting* or *forgetting factor* when computing $\mathbf{R}_n$ [Ref. 10]. The forgetting factor is a scalar value that may be designated as $\beta(n)$ where n is the iteration number of the point being estimated. The weighting factor $\beta(n)$ has the property that

$$0 < \beta(n) \leq 1, \quad n = 1, 2, \ldots, p \,, \tag{4.6}$$

where p is the total number of points being estimated. The purpose of $\beta(n)$ is to ensure that data in the distant past is 'forgotten' which will make it possible for the filter to adapt to statistical variations of the observed data when operating in a non-homogeneous environment. A commonly used form of the weighting factor is the *exponential weighting factor* which is defined as

$$\beta(n) = \lambda^{p-n}, \quad n = 1, 2, \ldots, p \,, \tag{4.7}$$

where λ is positive constant with a value less than 1. The quantity $1/(1-\lambda)$ can be considered as an approximate measure of the *memory* of the algorithm. When $\lambda = 1$ we have the ordinary method of least squares which corresponds to *infinite memory*. By this reasoning, we may introduce a *method of exponentially weighted least squares*,

where $\beta(n)$ is used as a weighting factor in the expression of the performance measure [Ref. 10]

$$\xi(n) = \sum_{n=1}^{p} \lambda^{p-n} |e(n)|^2 , \tag{4.8}$$

where $e(n)$ is the error defined by (2.12) at mask position (n_1, n_2).

To see how the forgetting factor is introduced in the recursion we must refer to the original definition of the normal equation

$$\mathbf{Ra} = \mathcal{E} , \tag{4.9}$$

where $\mathbf{a}$ is the vector of optimum parameters which results in the minimum value of the performance measure $\xi(n)$. The $P_1P_2 \times P_1P_2$ autocorrelation matrix in this case is defined by

$$\mathbf{R}_n = \sum_{n=1}^{p} \lambda^{p-n} \mathbf{x}_n \mathbf{x}_n^T . \tag{4.10}$$

We may now isolate the term of (4.10) that corresponds to $n = p$ from the rest of the summation on the right side of the expression to obtain

$$\mathbf{R}_n = \lambda \left[\sum_{n=1}^{p-1} \lambda^{p-1-n} \mathbf{x}_{n-1} \mathbf{x}_{n-1}^T \right] + \mathbf{x}_n \mathbf{x}_n^T . \tag{4.11}$$

The term inside the brackets on the right side of (4.11) is actually the sum of the previously computed autocorrelation matrices $\mathbf{R}(n-1)$, therefore we have the recursive relationship

$$\mathbf{R}_n = \lambda \mathbf{R}_{n-1} + \mathbf{x}_n \mathbf{x}_n^T . \tag{4.12}$$

The affect of $\beta(n)$ is apparent when (4.12) is expanded to give

$$\begin{aligned}
\mathbf{R}_1 &= \mathbf{x}_1 \mathbf{x}_1^T \\
\mathbf{R}_2 &= \lambda \mathbf{x}_1 \mathbf{x}_1^T + \mathbf{x}_2 \mathbf{x}_2^T \\
\mathbf{R}_3 &= \lambda^2 \mathbf{x}_1 \mathbf{x}_1^T + \lambda \mathbf{x}_2 \mathbf{x}_2^T + \mathbf{x}_3 \mathbf{x}_3^T \\
&\vdots \\
\mathbf{R}_n &= \lambda^{n-1} \mathbf{x}_1 \mathbf{x}_1^T + \lambda^{n-2} \mathbf{x}_2 \mathbf{x}_2^T + \ldots + \mathbf{x}_n \mathbf{x}_n^T
\end{aligned} \tag{4.13}$$

which clearly shows the decreasing contribution made by past data vectors when $\lambda < 1$.

The choice of a value for λ is dependent on the statistical nature of the data being processed. Generally, if the data is random or uncorrelated it is undesirable to use the past data to compute the current estimate, therefore a small value for λ would be the appropriate choice. Conversely, a value of λ close to 1 would be desirable for highly correlated data.

Since the statistical nature of the data is often not known *a priori* when using data-adaptive filters, it is difficult to choose an optimum value for the forgetting factor. A further modification of the recursive equation (4.12) has been implemented for the data-adaptive iterative algorithm to address this problem. This modification involves the application of a weighting factor to the current update of $\mathbf{R}_n$. This weighting factor depends on the index of iteration n such that its value approaches 1 as $n \longrightarrow \infty$. This biasing scheme causes the influence of the present data on the parameter estimates to increase with the number of iterations. The modified recursion is given by

$$\mathbf{R}_n = \lambda \mathbf{R}_{n-1} + \left(\frac{n-1}{n}\right) \mathbf{x}_n \mathbf{x}_n^T \,. \tag{4.14}$$

The weighting factor on the second term in (4.14) is added to provide a more gradual shift of influence to recent updates than would be possible with only the λ term present. This factor is not as important when a highly correlated signal is processed.

C. SUMMARY

The adaptive Toeplitz approximation algorithm can be used to estimate parameters for the same regions of support as in the fixed data case. Additionally, various schemes for pre-windowing or post-windowing the data may be employed to improve the estimates. Other factors affecting the performance of the algorithm include the

directional scheme for moving the filter over the data array and the criteria used to determine convergence of the parameter estimates. This algorithm has been used for spectral estimation, image noise cancellation, and line enhancement. Experimental results involving these applications are presented in the next two chapters.

V. 2–D AR SPECTRAL ESTIMATION

A. OVERVIEW

One application where the use of two-dimensional autoregressive modeling has proven to be particularly advantageous is that of spectral estimation. The 2–D AR model was derived in Chapter II. This model is used to find the 2–D AR power spectral estimate $\hat{P}_x(\omega_1, \omega_2)$ which is given by [Ref. 4]

$$\hat{P}_x(\omega_1, \omega_2) = |H(\omega_1, \omega_2)|^2 P_w \,, \tag{5.1}$$

where P_w is the power spectral density of the input and $H(\omega_1, \omega_2)$ is the transfer function of the 2–D AR model. The input is white noise with a constant power spectrum of amplitude σ_w^2. Therefore, we can write (5.1) as

$$\hat{P}_x(\omega_1, \omega_2) = \frac{\sigma_w^2}{|1 + \sum\sum_{(k_1,k_2 \in A)} a(k_1, k_2) e^{-j(\omega_1 k_1 + \omega_2 k_2)}|^2} \,. \tag{5.2}$$

The key to finding the 2–D AR power spectral estimate is determining the the parameters $a(k_1, k_2)$. The next section provides results of experimentation with the data-adaptive iterative algorithm as used to find the parameter estimates of a 2–D signal consisting of a sum of sinusoids in additive white noise. Results using the direct inversion method and fixed data iterative Toeplitz approximation are provided for comparison.

B. EXPERIMENTAL RESULTS

The performance of the algorithms discussed in the previous chapters is compared by examining the estimated spectral densities computed from the parameters produced by each method. Each algorithm has been used to estimate the parameters

of the sinusoidal input array generated by

$$\begin{aligned} x(n_1, n_2) &= A_1 \cos(2\pi f_{11} n_1 + 2\pi f_{12} n_2) \\ &\quad + A_2 \cos(2\pi f_{21} n_1 + 2\pi f_{22} n_2) + w(n_1, n_2) \end{aligned} \tag{5.3}$$

where amplitudes $A_1 = A_2 = \sqrt{2}$ and f_{ij} are normalized frequencies in the range $(0 \leq f_{ij} \leq 0.5)$. The frequencies chosen are $f_{11} = f_{12} = 0.167$ and $f_{21} = f_{22} = 0.333$. Additionally, the data-adaptive algorithm is tested with off-set frequencies to evaluate its peformance for that case. The noise term $w(n_1, n_2)$ is zero mean and gaussian with the variance σ_w^2 scaled give a desired signal-to-noise ratio (SNR). The SNR (in dB) is defined as [Ref. 4]

$$\text{SNR} = 10 \log_{10} \sum_{i=1}^{N} \frac{A_i^2}{2\sigma_w^2} , \tag{5.4}$$

where N is the number of sinusoids present and A is the amplitude term. The SNR is varied by holding A_i constant and adjusting the value of σ_w^2. The algorithm's performance using a 3×3 filter mask is evaluated for various sizes of input arrays and the common regions of support, including combined-quadrant (CQ). The results obtained using the data-adaptive algorithm are provided for the pre-windowed and covariance methods. The data-adaptive algorithm is tested with SNR's of 10 dB and 0 dB. It should be noted that the surface plots in all figures have been rotated 90 degrees to show the separation of the spectral peaks more clearly. Contour plots are provided to better judge the accuracy of the estimates. The axes of the contour plots range from 0-30 which is taken from a 32×32 frequency domain array representing a quadrant of a 64×64 point 2–D FFT output. The 32×32 array covers the frequency range from 0 to π.

1. Estimates by Direct Inversion

The first case examined is that involving direct inversion of the autocorrelation matrix to find the least squares AR parameters from the normal equations. An

8×8 input array was used with 1^{st} quadrant quarter-plane support. The resulting spectral density estimate is shown in Figure 5.1. This plot shows spectral peaks at

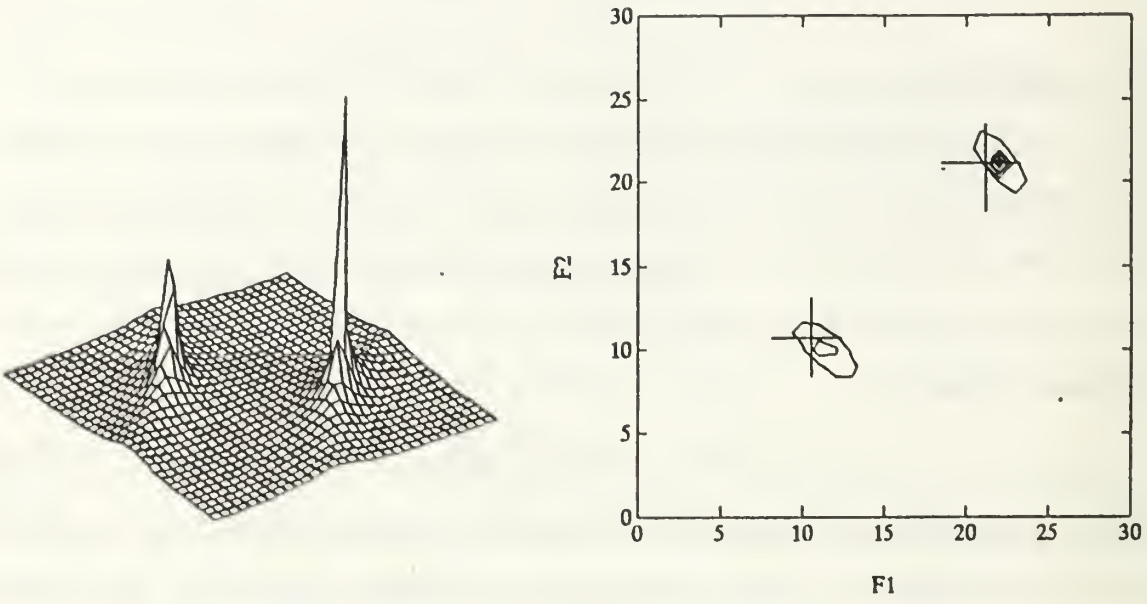

Figure 5.1: Spectral estimate using direct matrix inversion; SNR=10 dB.

21.3 and 10.7 on the F1 and F2 axis respectively, which correspond to the normalized frquencies $f_{11} = f_{12} = 0.167$ and $f_{21} = f_{22} = 0.333$ of the input data. The true locations of the frequencies are indicated by crosses.

2. Spectral Estimates using the Iterative Method for Fixed Data

a. Quarter-Plane Support

Figure 5.2 shows the spectral density of $x(n_1, n_2)$ as estimated using the fixed data iterative method with first quadrant QP support. As in the previous case the frequencies estimated are very close to the true frequencies. It is interesting to note however, that contrary to what might be expected, the quality of the estimate appears to degrade as the number of iterations is increased. The estimate produced after one iteration is clearly superior to the estimate obtained after ten iterations.

This phenomenon is consistent with that experienced in the 1-D case where the iterative solution would begin to diverge after a certain number of iterations in cases with small data sets [Ref. 12].

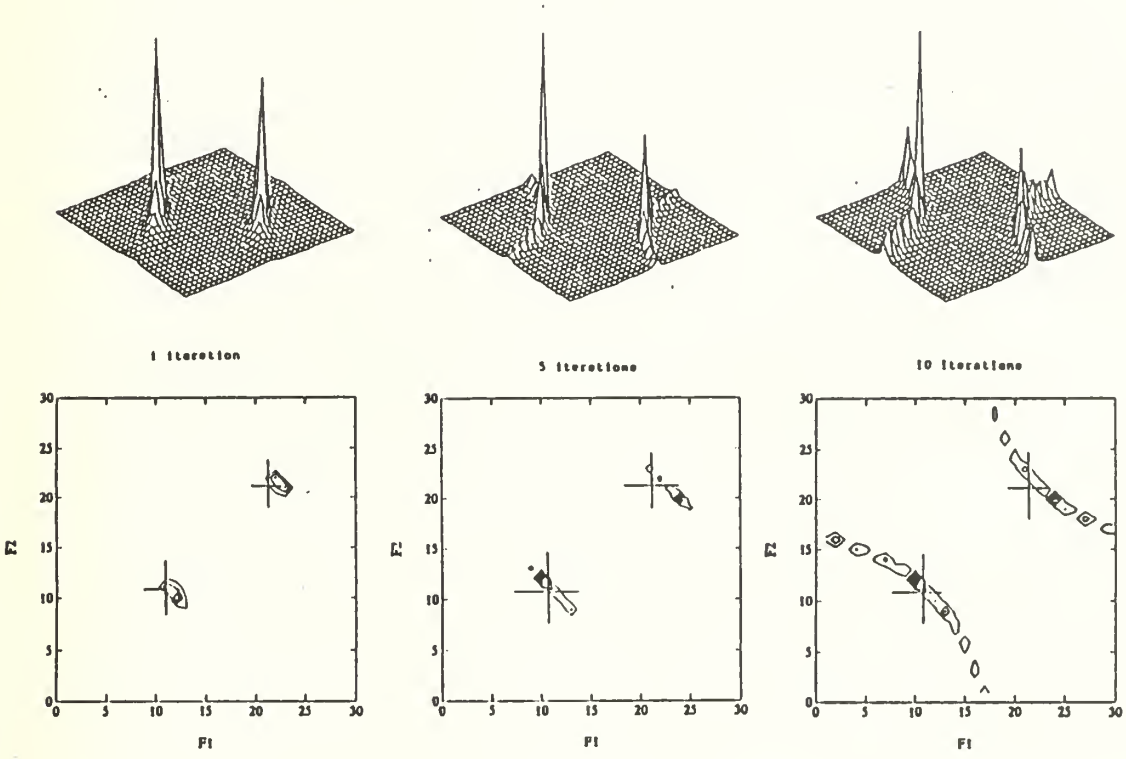

Figure 5.2: Fixed Data, 1st Quadrant QP Support, SNR=10 dB.

Results using second quadrant QP support proved to be generally better for this placement of the sinusoids than those obtained from the first quadrant. An example is provided in Figure 5.3.

b. Combined Quadrant Support

Spectral density estimates using QP support demonstrate a tendency to distort elliptically. This distortion is evident in the first quadrant QP case shown in Figure 5.2. A method to compensate for this elongation of the spectral peaks

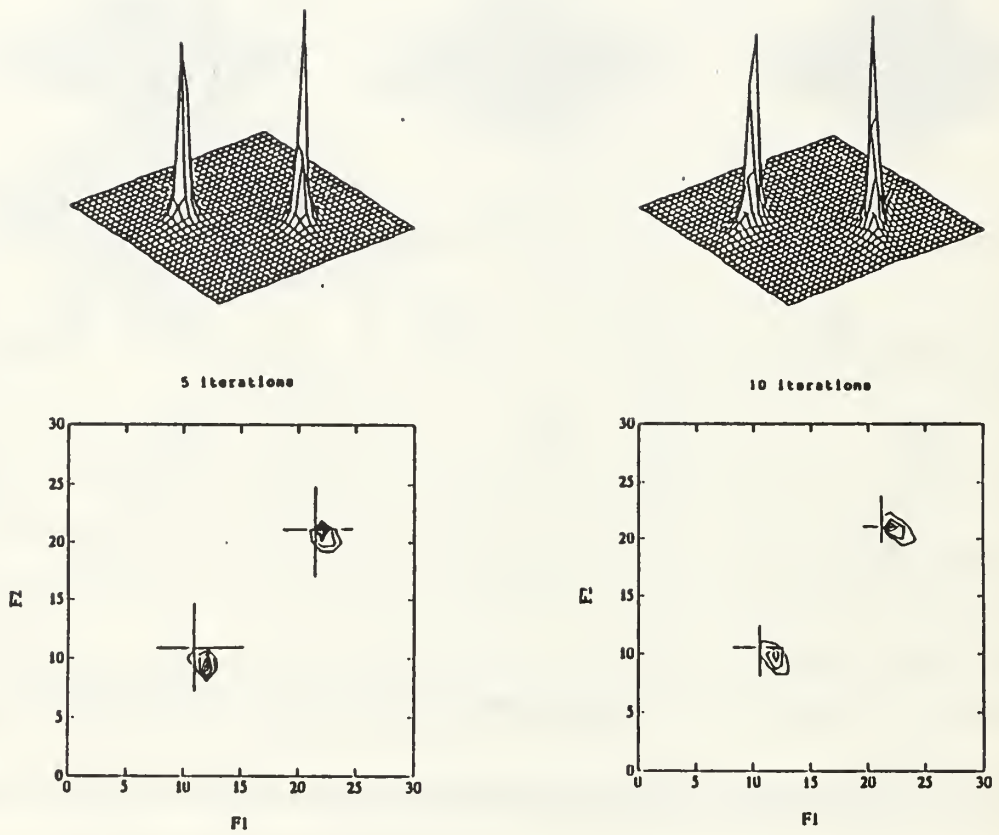

Figure 5.3: Fixed Data, 2nd Quadrant QP Support, SNR=10 dB.

involves combining the first and second quadrant estimates [Ref. 13] to give

$$\hat{P}_x = \frac{2\sigma_w^2}{\left[\hat{P}_{q1}^{-1} + \hat{P}_{q2}^{-1}\right]}. \tag{5.5}$$

The plots in Figure 5.4 indicate that this method yields accurate estimates. However, despite some improvement, the elliptical elongation at the base of the spectral peaks resulting from the influence of the first quadrant estimate is still quite noticeable.

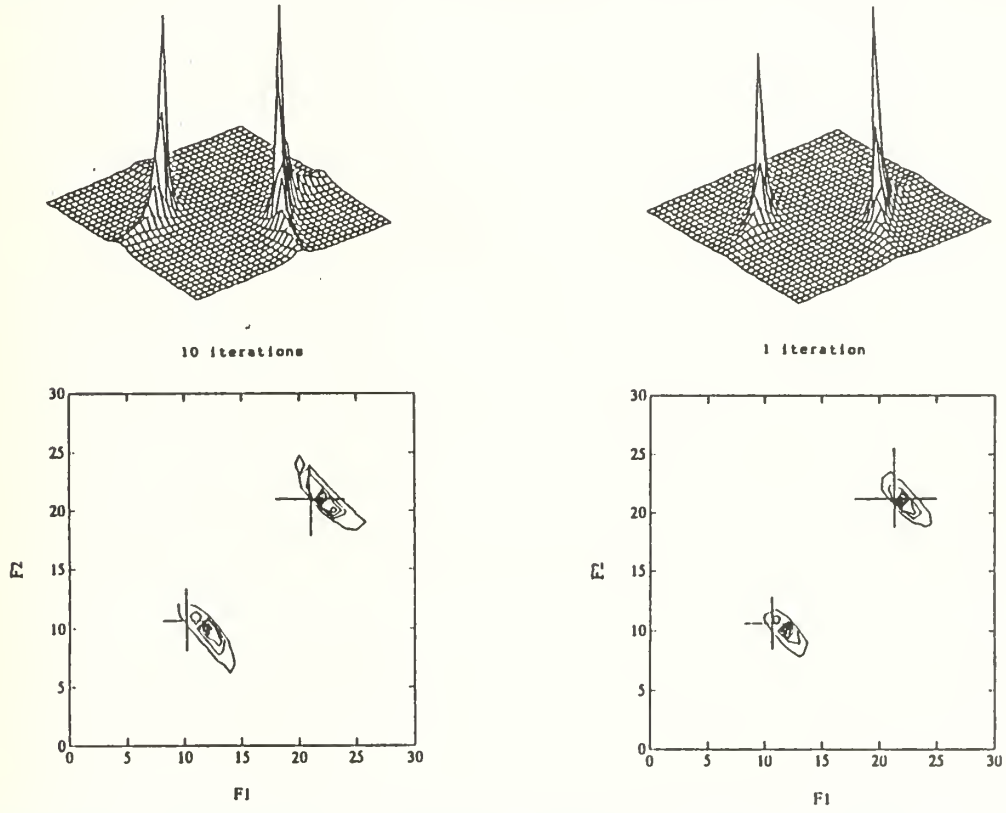

Figure 5.4: Fixed Data, Combined Quadrant Support, SNR=10 dB.

3. Spectral Estimates using the Data-Adaptive Iterative Method

a. Pre-Windowed Input Arrays

The input arrays were pre-windowed by adding $P_1 - 1$ rows and columns of zeros to the top and left side of the array, where P_1 is the dimension of the filter mask. The purpose of pre-windowing the data is to compute the parameter estimates using all the input data. Figure 5.5 shows the spectral estimates

for 1^{st}, 2^{nd} and CQ support with an 8×8 input array. In Figure 5.6 we present the

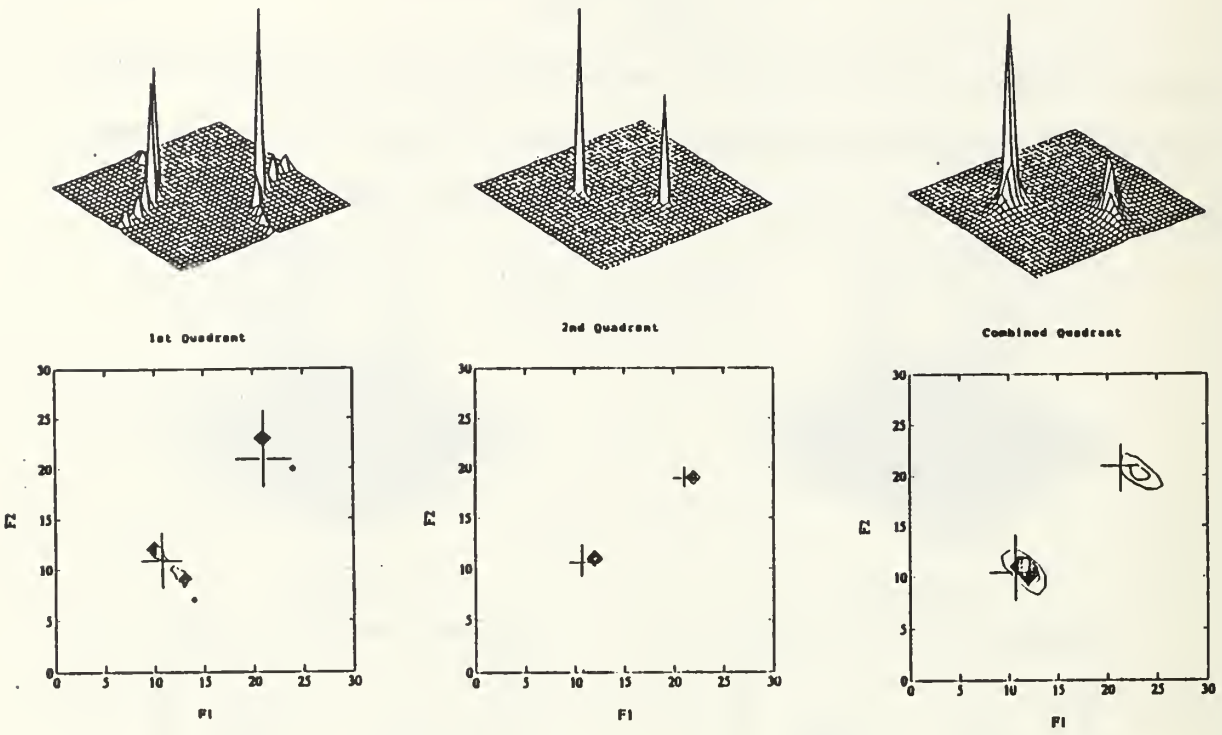

Figure 5.5: 8×8 input array (pre-windowed), SNR=10 dB.

spectral estimates when only 16 data points (4×4 array) were available. The spectral estimates obtained from the 8×8 input array were quite acceptable, but some bias is evident in the first quadrant estimate which is carried over to the CQ estimate. The estimates obtained from the 4×4 array demonstrate the value of computing the CQ estimate. In this case the true signal frequencies were unrecognizable until the combined quadrant estimate was obtained.

A final test of the pre-windowed algorithm consisted of running the filter mask only half way through an 8×8 input. The estimates are computed from the upper 4×8 section of the observed data. The result of this experiment is given in Figure 5.7. These results are similar to that of the 4×4 case and serve to reinforce the

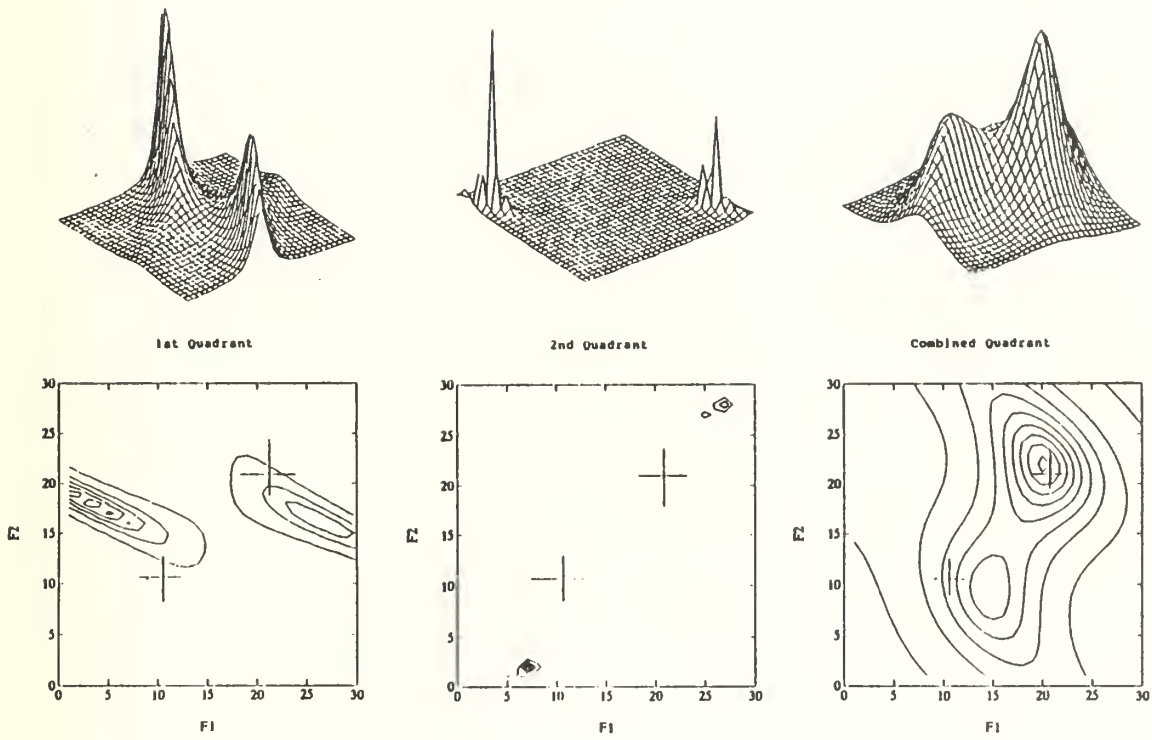

Figure 5.6: 4×4 input array (pre-windowed), SNR=10 dB.

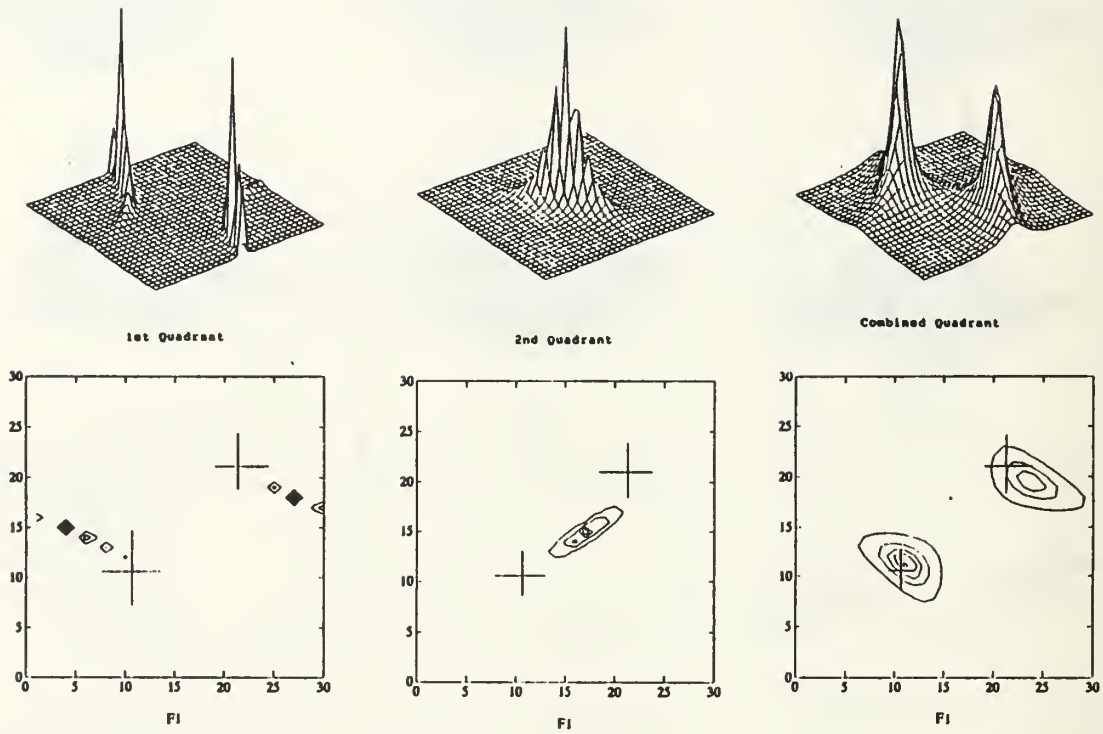

Figure 5.7: 4×8 **portion of the input array (pre-windowed), SNR=10 dB.**

importance of the Combined Quadrant technique for 2-D AR spectrum estimation.

b. Estimates using the Covariance Method

As explained in section B of Chapter II the covariance method uses only the observed data. As a result, $P_1 - 1$ rows and columns of the input array are not available when computing the parameter estimates. However, the experimental results obtained in this work indicate that the covariance method achieves better spectral peak resolution. Additionally, less bias is observed in the spectral estimates. This characteristic of the covariance method is evident in Fig 5.8. In this case the

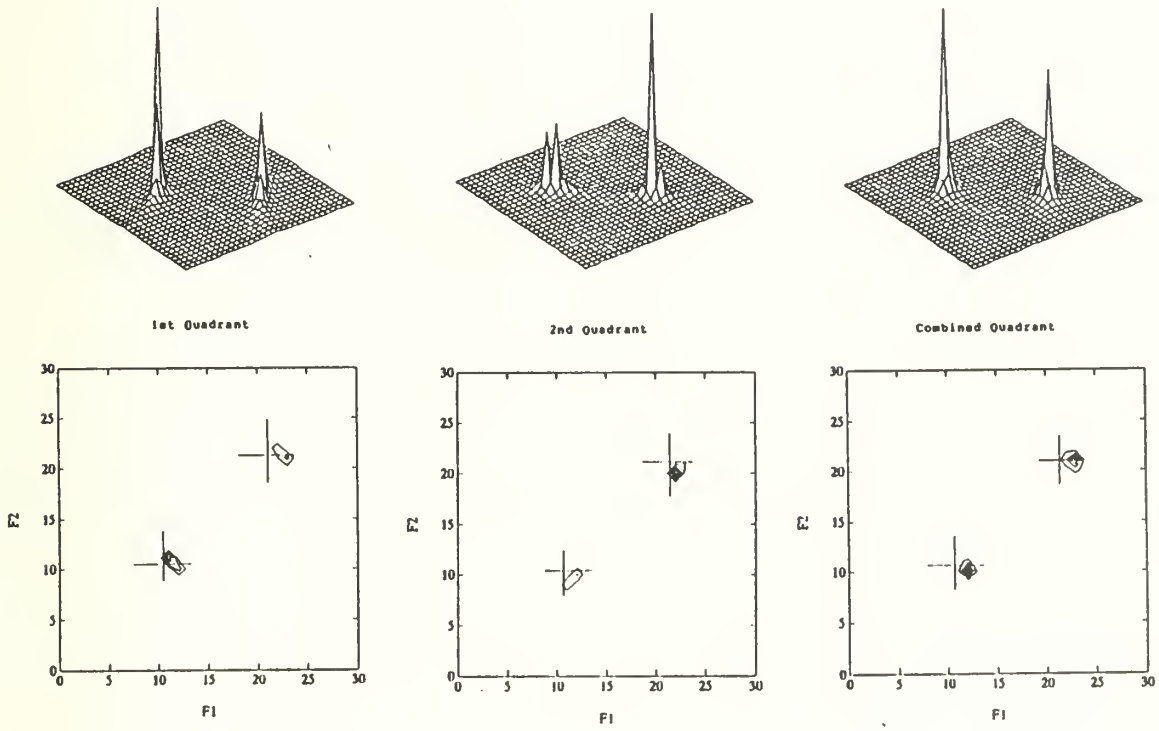

Figure 5.8: 8×8 input array (Covariance Method), SNR=10 dB.

best spectral estimate is clearly provided by the Combined Quadrant method. The bias present in the pre-windowed first quadrant support example has been completely eliminated by using the covariance method.

The next example demonstrates a unique characteristic of the data-adaptive iterative Toeplitz approximation algorithm. In this example the spectral estimate is computed using a 3×3 filter mask with a 4×4 input array. This means that only *four* elements of the input data are available to the filter to determine the parameter estimates. This result is of particular interest because the correlation matrix is actually rank deficient, and the problem cannot be solved by direct inversion. Nevertheless, the iterative Toeplitz approximation method does provide a means to compute the estimate. The results are given in Figure 5.9. Although the spectral

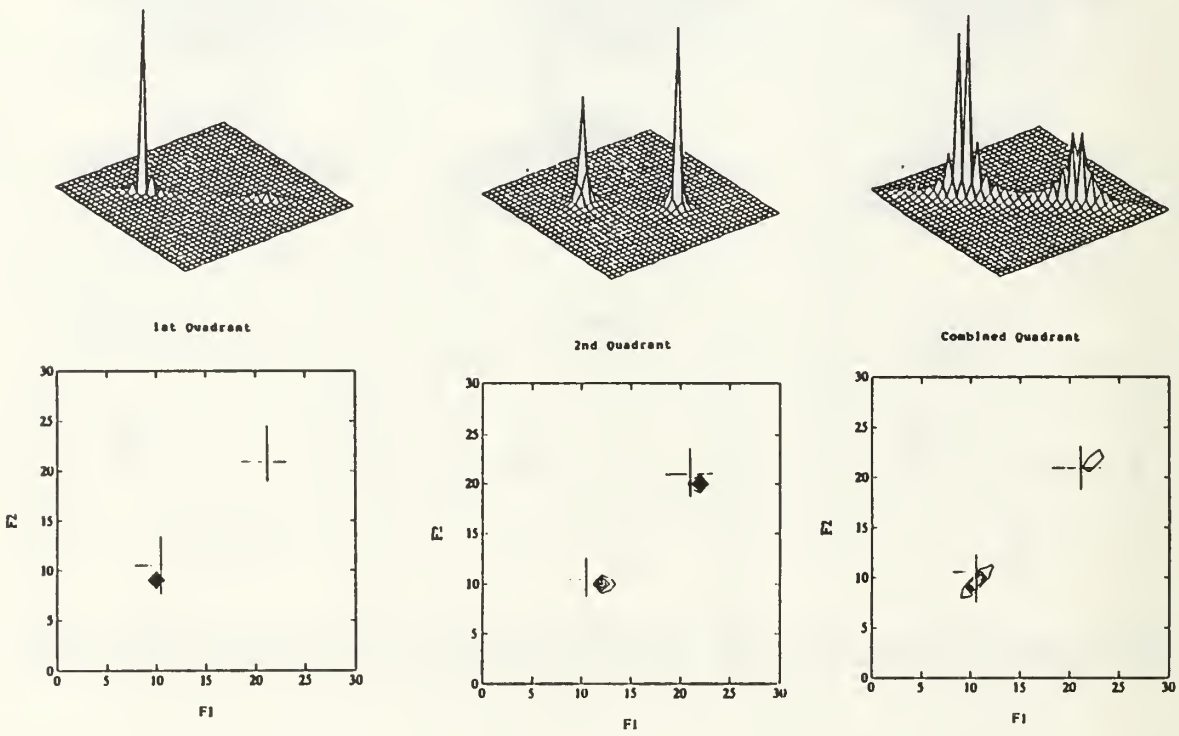

Figure 5.9: 4×4 input array (Covariance Method), SNR=10 dB.

estimates for the first quadrant and Combined Quadrants are somewhat weak, the second quadrant estimate exhibits remarkable resolution and accuracy.

As a further test of the algorithm, a signal with offset frequencies

$f_{11} = 0.250$, $f_{12} = 0.125$, $f_{21} = 0.300$, and $f_{22} = 0.400$ is generated and processed. The spectral estimates resulting from the computed model parameters are plotted in Figure 5.10. For this case, the best spectral estimate was obtained using first

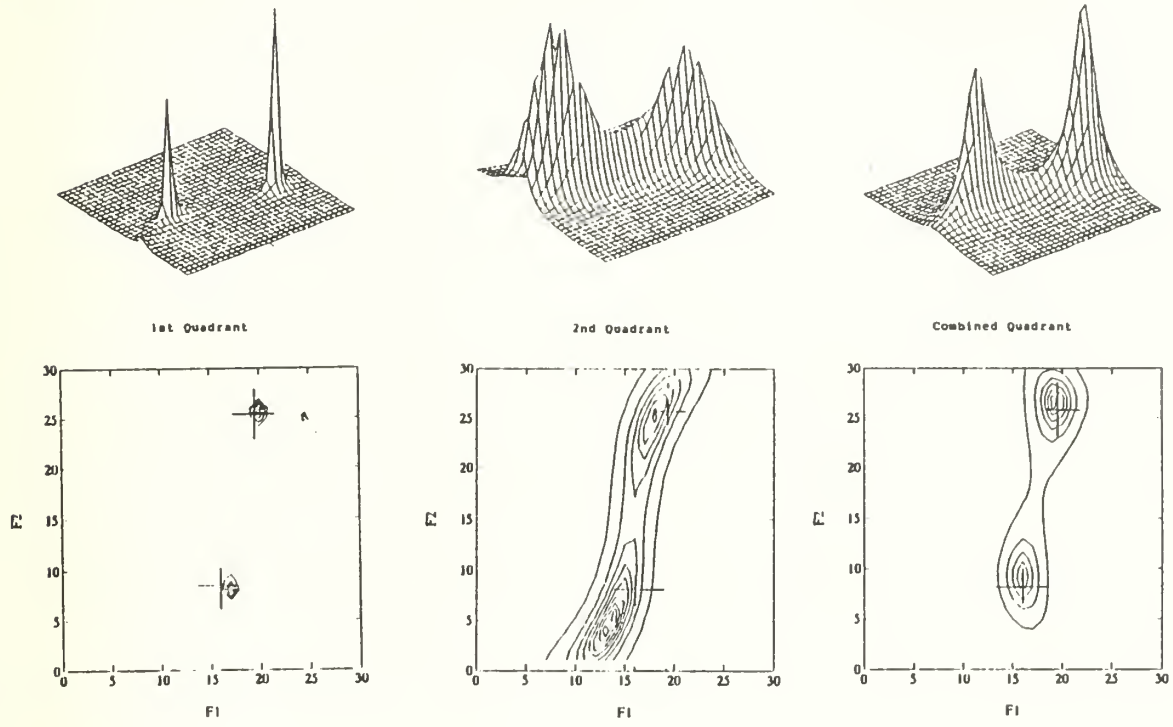

Figure 5.10: 8×8 input array, offset frequencies, (Covariance Method), SNR=10 dB.

quadrant suppport.

The spectral estimates for a signal-to-noise ratio of 0 dB is ploted in Figure 5.11. The accuracy of the estimates is noticeably degraded in this case. However, a reasonable indication of the signal frequency content can be seen when the combined-quadrant method is used.

The final case using NSHP support is included for completeness. The filter mask had dimensions $P_1 = P_2 = 4$ and $L_2 = -2$. The parameters were estimated for a 16×16 input array. The NSHP spectral estimate is plotted in Figure 5.12. The

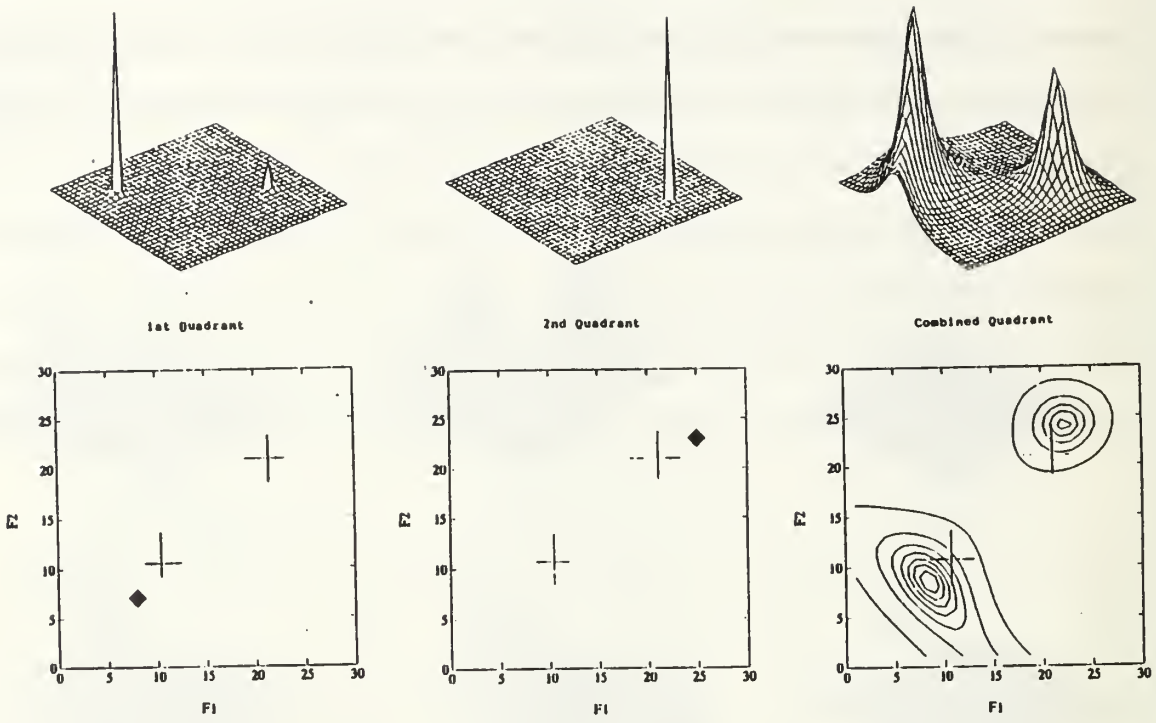

Figure 5.11: 8×8 input array (Covariance Method), SNR=0dB.

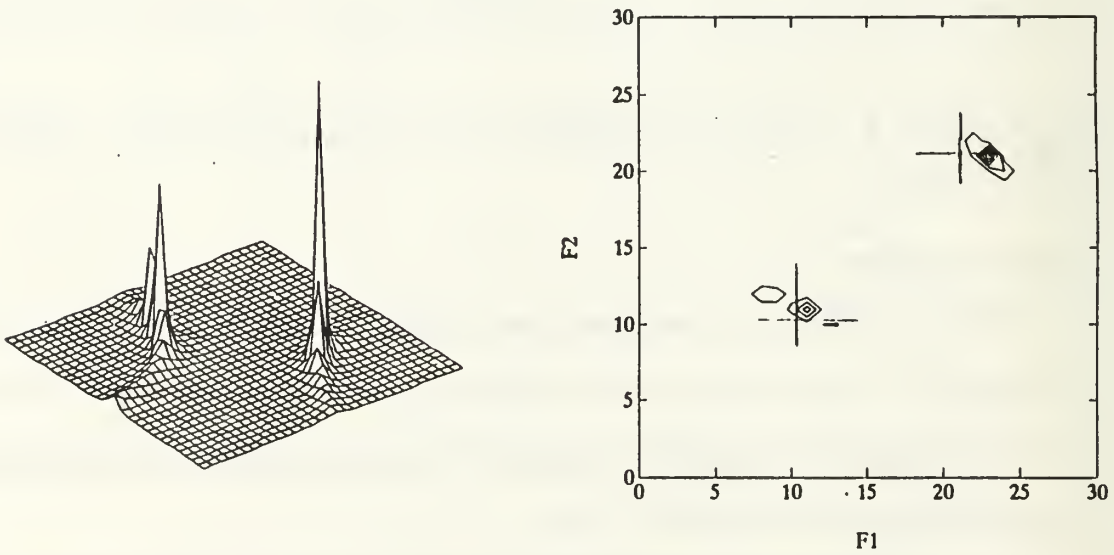

Figure 5.12: 16×16 input array (Covariance Method), SNR=10 dB; non-symmetric half-plane support.

resulting spectral estimate shown in Figure 5.12 is quite accurate, but this comes at the cost of greater complexity for the filter. NSHP support may be used as an alternative to combined-quadrant support in some cases.

C. SUMMARY

The data-adaptive iterative Toeplitz approximation algorithm has proven to be a viable means for 2–D AR spectrum estimation. Numerous factors must be considered when using this algorithm. In particular, the filter mask size should be chosen to correspond to the model order of the signal being processed. Although our overall experience has found that that CQ support produced the best results, no region of support seemed to have a clear advantage over the others for all cases. This indicates that the optimum region of support must be evaluated for the specific application that the algorithm is implemented for.

VI. RESTORATION OF IMAGES

A. OVERVIEW

The characteristics of typical images vary greatly from one region of the image to the next. For example, a region of an image containing a crowd or a building may have detailed variations in intensity, while another region representing the sky in the background will be essentially uniform in intensity. Two-dimensional data-adaptive filtering is particularly suited for processing data of this nature. The idea of adapting the processing to the local characteristics of an image is advantageous for many image processing applications, including image enhancement and restoration.

There are two approaches to adaptive signal processing. The first approach involves adapting the filter output for each pixel. This is known as *pixel-by-pixel processing* [Ref. 4]. In this scheme, the processing method is based on the local characteristics of the image, degradation, and any other pertinent information contained in the pixel's neighborhood region. This approach offers the greatest degree of flexiblity to the adaptive process, but has the highest computational complexity.

When a more computationally efficient method is desired, a second approach known as *subimage-by-subimage* or *block-by-block* processing can be used [Ref. 4]. In this approach, the image array is divided into subimages and space-invariant techniques are used to process the data. This method can result in some discontinuities being present in the processed image. The size of the subimages directly affects the quality of the output image. The block-by-block method provides a faster means of processing, however, and may be desirable in some applications.

In the next two sections of this chapter the data-adaptive Toeplitz approxima-

tion algorithm is implemented in image noise cancellation and line enhancer modes. For both of these applications the algorithm has been modified to perform pixel-by-pixel processing.

B. IMAGE NOISE CANCELLATION

The usual method of finding the estimate of a signal in noise is to pass the corrupted signal through a system that serves to suppress the noise while leaving the desired signal relatively unchanged. The noise canceler developed by Widrow [Ref. 14] is an example of such a system. Adaptive noise cancellation is a variation of optimal filtering that can be used for image restoration to some advantage. In particular, when a recieved image is obscured by noise or another image transmitted from another location as shown in Figure 6.1, an adaptive noise canceler can be employed to extract the desired image from the composite signal. The noise canceler

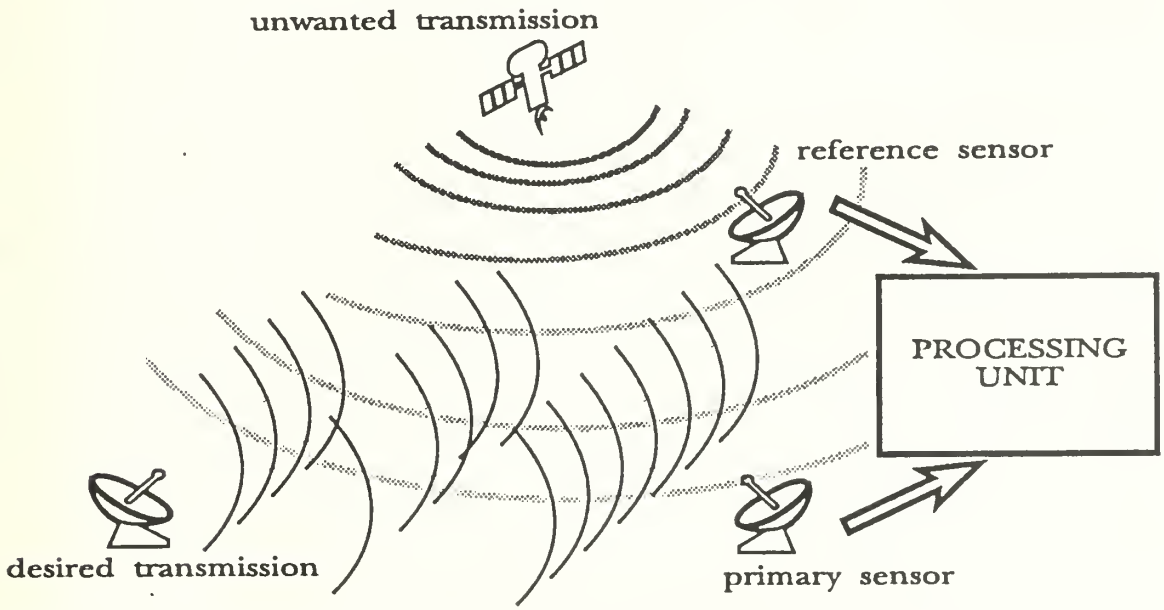

Figure 6.1: A possible scenario for application of a noise canceler as used for image restoration.

makes use of an auxiliary or reference input derived from one or more sensors located

at points in the noise field where the desired signal is weak or undetectable. This input is filtered and subtracted from the primary input containing both desired and unwanted signals. A block diagram of the noise canceler is shown in Figure 6.2. The reference signal and the undesired part of the primary input are correlated. For this reason, the unwanted signal or noise is eliminated or attenuated by cancellation.

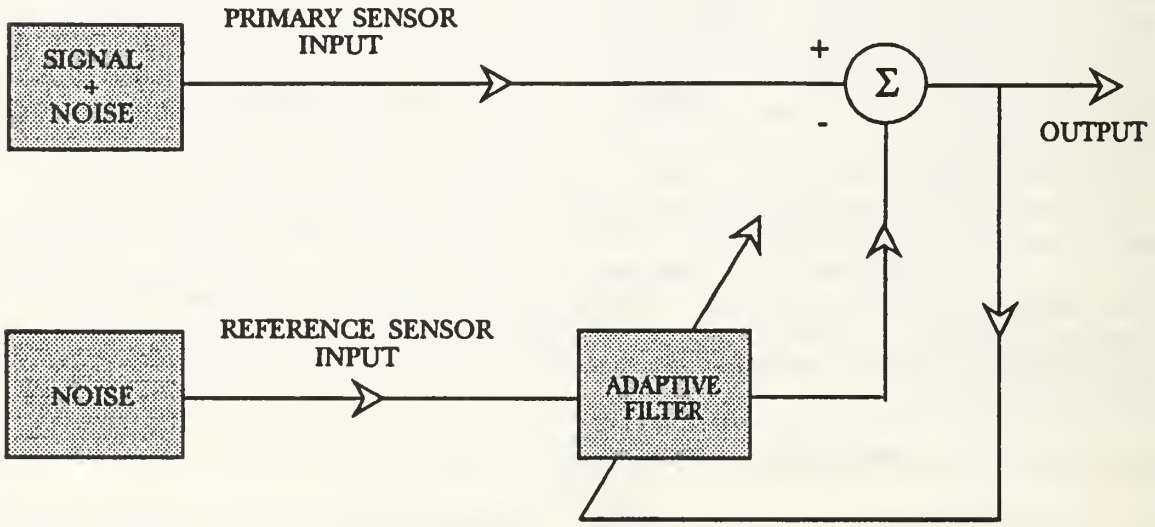

Figure 6.2: Adaptive Noise Canceler Block Diagram.

To evaluate the performance of the data-adaptive iterative Toeplitz approximation algorithm used in a noise canceler configuration, a 256×256 composite image array was created from two distinct images. This corrupted image is shown in Figure 6.3. The algorithm described in Chapter IV was modified to include a cross-correlation term given by

$$\mathbf{r}_n = \lambda \mathbf{r}_{n-1} + \left(\frac{n-1}{n}\right) d_n \mathbf{x}_n \tag{6.1}$$

where d_n is the element of the composite signal corresponding to the n^{th} element of the reference signal being processed by the filter. This cross-correlation term is used to form the initial parameter estimate for each pixel being processed.

Figure 6.3: Corrupted image array provided as primary input to noise canceler.

The resulting error $e(n_1, n_2)$ computed at the output of the system becomes the restored image. This relationship is given by

$$e(n_1, n_2) = [d(n_1, n_2) + u(n_1, n_2)] - \hat{u}(n_1, n_2) \,, \tag{6.2}$$

where d is the desired image, u is the unwanted image or noise and $\hat{u}$ is the filtered estimate of u.

In the simulation the noise signal added to the desired image is scaled and lowpass filtered to simulate the effect of having the primary and reference sensors physically separated. The output of the noise canceler is shown in Figure 6.4. The unwanted image has been noticeably attenuated and the desired image is clearly visible. The original images are provided in Figure 6.5 as a reference for evaluation of the algorithm's performance.

C. ADAPTIVE LINE ENHANCER

A special case of adaptive noise canceling occurs when only the signal that is corrupted by noise is available. In this case the recieved signal is delayed and used as the reference signal. The reference signal may be expressed as $y(n_1, n_2) = x(n_1 - \delta, n_2)$, where δ is the amount of spatial delay used. The block diagram for this system is

Figure 6.4: Processed image array after noise cancellation.

Figure 6.5: The original images - undesired and desired.

shown in Figure 6.6. The main function of the delay parameter δ is to remove any

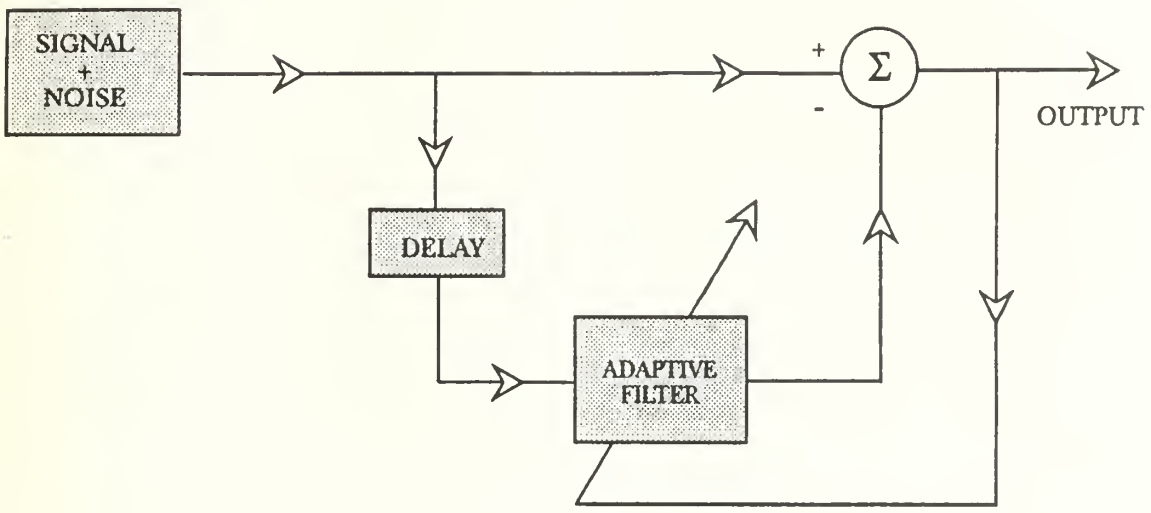

Figure 6.6: The adaptive line enhancer block diagram.

correlation that may exist between the noise component in the original input signal and the noise component of the delayed adaptive filter input [Ref. 10]. The filter will respond by cancelling any components of the main signal $x(n_1, n_2)$ that are in any way correlated with the secondary signal $y(n_1, n_2)$ [Ref. 11]. The remaining noise component at the output of the filter is then subtracted from main signal resulting in the removal of the additive noise in the signal. In general, the delay δ should be chosen such that it is approximately half the length of the correlation sequence [Ref. 11]. The input image and output of the adaptive line enhancer are shown in Figure 6.7.

D. SUMMARY

In this chapter, it has been demonstrated that the data adaptive Toeplitz approximation algorithm may be used in image processing applications. While pixel-by-pixel processing was used in the above examples, it may be desirable to combine this

Figure 6.7: Input and output of the data-adaptive line enhancer.

approach with block-by-block processing to obtain better results. This would entail dividing the image into smaller blocks before processing and then recombining the processed blocks. Additionally, reprocessing of the array estimates may be a viable means of improving image quality.

VII. CONCLUSIONS

The data-adaptive iterative Toeplitz approximation algorithm is readily implemented from the iterative methods previously used for fixed data-arrays. The experimental results indicate that this algorithm is well-suited for a wide variety of 2–D signal processing applications. Practically speaking, the iterative method can be successfully implemented for any application where the use of Wiener filter based adaptive filters has been successful. While more extensive evaluation is necessary before this method can be said to have a clear advantage over other methods, there is enough evidence of its capabilities to warrant further investigation in this area.

A. PERFORMANCE EVALUATION SUMMARY

As stated earlier, autoregressive model parameters may used to estimate the spectral content of 2–D arrays with high resolution. In experiments the performance of the data-adaptive iterative algorithm matched that of the direct inversion method, and in some cases, improved upon the performance of the fixed data iterative method. This improvement may be due to the way in which the iteration is carried out in the adaptive case versus the fixed case. In the fixed case the iteration is carried out using a correlation matrix formed from all the data at once. In the data-adaptive case the correlation matrix is formed recursively with new data incorporated at each iteration. Of particular note, was the ability of the data-adaptive algorithm to estimate frequencies using very small data sets and its ability to produce an estimate even when the correlation matrix is not of full rank . Additionally, its performance in a high noise environment was noteworthy.

The algorithm also proved to be a viable method for data-adaptive image

restoration. The nature of this application differs greatly from that of spectral estimation in that the data sets are much larger and generally less correlated. Encouraging results were obtained for the data-adaptive noise canceler and line enhancer with only minor modifications of the same algorithm that was used for the spectrum estimates.

B. RECOMMENDATIONS FOR FUTURE STUDY

The primary objective of this thesis was to develop the data-adaptive iterative Toeplitz algorithm and obtain experimental results to evaluate its potential for use in applications that involve 2–D AR parameter estimation. There are many aspects regarding this work that require more in-depth study and several applications that may benefit in the use of this algorithm. In particular, more detailed analysis of model orders and their relation to the algorithm's ability to form spectral estimates from very small input arrays needs to be carried out. The use of singular value decomposition (SVD) and eigenvalue analysis of the observed data may provide some insights regarding the performance of the iterative algorithm. Also, further investigation as to why one region of support (ROS) provides greater resolution than another, may yield information that would permit the choice of an optimum ROS prior to processing the data. Further analysis for choosing values of the forgetting factor λ and how it relates to the statistical nature of the data, is desirable as well. This may lead to a scheme for on-line adjustment of λ, which would be particularly advantageous when processing images. Other future work could explore the use of different data windowing schemes and data-ordering schemes for passing the filter mask over the data. One possible method that has not been investigated is an expanding square starting at one corner of the array. This method seems well suited for block-by-block image processing. Finally, the data-adaptive iterative Toeplitz approximation algorithm has the potential for use in non-linear applications, such as the identification of Volterra

Thesis
E56644 Eremic
c.1 Iterative methods for estimation of 2-D AR parameters using a data-adaptive Toeplitz approximation algorithm.